

Please type a plus sign (+) inside this box → ☐

PTO/SB/50 (4/98)  
Approved for use through 09/30/2000. OMB 0651-0033  
Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE  
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

## REISSUE PATENT APPLICATION TRANSMITTAL

Address to:

Assistant Commissioner for Patents  
Box Patent Application  
Washington, DC 20231

Attorney Docket No.	1088.300
First Named Inventor	BICKENS
Original Patent Number	5,806,063
Original Patent Issue Date (Month/Day/Year)	September 8, 1998
Express Mail Label No.	

APPLICATION FOR REISSUE OF:  
(check applicable box)



Utility Patent



Design Patent



Plant Patent

### APPLICATION ELEMENTS

- ☐ \* Fee Transmittal Form (PTO/SB/56)  
(Submit an original, and a duplicate for fee processing)
- ☐ Specification and Claims (amended, if appropriate)
- ☐ Drawing(s) (proposed amendments, if appropriate)
- ☐ Reissue Oath / Declaration (original or copy)  
(37 C.F.R. § 1.175)(PTO/SB/51 or 52)
- Original U.S. Patent  
☐ Offer to Surrender Original Patent (37 C.F.R. § 1.178)  
(PTO/SB/53 or PTO/SB/54)  
or  
☐ Ribboned Original Patent Grant  
☐ Affidavit / Declaration of Loss (PTO/SB/55)
- Original U.S. Patent currently assigned?  
☒ Yes ☐ No  
(If Yes, check applicable box(es)).  
☒ Written Consent of all Assignees (PTO/SB/53 or 54)  
☐ 37 C.F.R. § 3.73(b) Statement ☒ Power of Attorney

### ACCOMPANYING APPLICATION PARTS

- ☐ Foreign Priority Claim (35 U.S.C. 119)  
(if applicable)
- ☒ Information Disclosure Statement (IDS)/PTO-1449 ☒ Copies of IDS Citations
- ☐ English Translation of Reissue Oath/Declaration  
(if applicable)  
\* Small Entity ☐ Statement filed in prior application,  
10. ☒ Statement(s) ☐ Status still proper and desired  
(PTO/SB/09-12)
- ☒ Preliminary Amendment
- ☒ Return Receipt Postcard (MPEP 503)  
(Should be specifically itemized)
- ☐ Other: \_\_\_\_\_

\* NOTE FOR ITEMS 1 & 10: IN ORDER TO BE ENTITLED TO PAY  
SMALL ENTITY FEES, A SMALL ENTITY STATEMENT IS REQUIRED  
(37 C.F.R. § 1.27), EXCEPT IF ONE FILED IN A PRIOR APPLICATION  
IS RELIED UPON (37 C.F.R. § 1.28).

### 14. CORRESPONDENCE ADDRESS

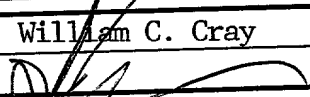
☐ Customer Number or Bar Code Label

022853

or ☐ Correspondence address below

(Insert Customer No. or Attach bar code label here)

Name				
Address				
City	State	Zip Code		
Country	Telephone	Fax		

NAME (Print/Type)	William C. Cray	Registration No. (Attorney/Agent)	27627
Signature		Date	2/23/00

Burden Hour Statement: This form is estimated to take 0.2 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Box Patent Application, Washington, DC 20231.

02-25-00

02/23/00  
JC644 U.S. PTO



384 FOREST AVENUE SUITE 13  
LAGUNA BEACH, CALIFORNIA 92651  
TEL 949.497.7676  
FAX 949.497.7679  
www.lagunalaw.com

A  
REIS  
JC598 U.S. PTO  
09/512592  
02/23/00

Patent ♦ Trademark ♦ Copyright ♦  
Trade Dress ♦ and Related Litigation

February 23, 2000

Via Express Mail EE87496H93 US

Box: Patent Application  
Assistant Commissioner for Patents  
Washington, DC 20231

Dear Commissioner:

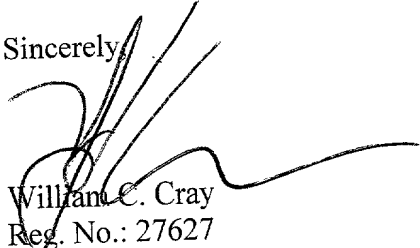
Enclosed a Reissue Application to reissue United States Patent No. 5,806,063  
Issued to Dickens on September 8, 1998  
Attorney Docket No. 2039-301

This submission comprises:

- Reissue Application Specification and Drawing
- Reissue Patent Application Transmittal
- Reissue Application Declaration By the Assignee
- Reissue Application By the Assignee, Offer to Surrender Patent
- 37 CFR §3.73 Declaration
- Statement Claiming Small Entity Status
- Reissue Application Fee Transmittal Form
- Preliminary Amendment
- Information Disclosure Statement
- Certificate of Mailing By Express Mail
- Check in the Amount of \$1,641.00
- Self addressed postcard

If you have any questions, please do not hesitate to contact me.

Sincerely,

  
William C. Cray  
Reg. No.: 27627

WCC/ns  
Enclosures  
cc: JEH  
WEL

REISSUE APPLICATION BY THE ASSIGNEE, OFFER TO SURRENDER PATENT		Docket Number (Optional)
This is part of the application for a reissue patent based on the original patent identified below.		
Name of Patentee(s): DICKENS		
Patent Number 5,806,063	Date Patent Issued September 8, 1998	
Title of Invention Date Formatting and Sorting For Dates Spanning the Turn of the Century		
<u>DICKENS-SOEDER2000</u> is the assignee of the entire interest in the original patent.		
I offer to surrender the original patent.		
<input type="checkbox"/> A certificate under 37 CFR 3.73(b) is attached.		
I am authorized to act on behalf of the assignee.		
I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under 18 U.S.C. 1001 and that such willful false statements may jeopardize the validity of the application, any patent issued thereon, or any patent to which this declaration is directed.		
Name of assignee DICKENS-SOEDER2000		
Signature of person signing for assignee <i>Bruce M. Dickens</i>		Date 2/22/00
Typed or printed name and title of person signing for assignee Bruce M. Dickens		

Burden Hour Statement: This form is estimated to take 0.1 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Washington, DC 20231.

**IN THE UNITED STATES PATENT  
AND TRADEMARK OFFICE**

<b>Reissue Application No.</b>	)	<b>Dated:</b> February 22, 2000
	)	
<b>United States Patent No. 5,806,063</b>	)	<b>Group Art Unit:</b> 2771
	)	
<b>Issued: September 8, 1998</b>	)	<b>Examiner:</b> Wayne Amsbury
	)	
<b>Applicant: Dickens-Soeder2000, LLC</b>	)	<b>Attorney Docket No.:</b> 2039-301
	)	

---

**Certificate Under 37 C.F.R. §3.73**

I Bruce M. Dickens as a Member and Manager of Dickens-Soeder2000, LLC, a Limited Liability Company created under the laws of the State of California, the owner of all of the right title and interest in the above referenced patent do hereby certify that Dickens-Soeder2000 is the owner of all of the right title and interest in and to the above referenced patent by virtue of an assignment from the assignee of record listed on the above referenced patent, McDonnell Douglas Corporation to Bruce Dickens dated April 2, 1999, which was submitted for recordation at the United States Patent and Trademark Office ("USPTO") on November 19, 1999, as shown by the attached documentation, and an assignment from Bruce Dickens and Karen Dickens, husband and wife and residents of the State of California, a community property state, to Dickens 2000, LLC, a Limited Liability Company created under the laws of the State of California, which was submitted for recordation in the USPTO on February 18, 2000, as shown by the attached, and from Dickens 2000 to Dickens-Soeder2000, LLC, dated February 18, 2000, and submitted for recordation at the USPTO on February 18, 2000, as shown by the attached. I hereby declare that the statements made herein by me of my own knowledge are true and that statements made by me on information and belief are believed to be true, and further that these statements are made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under 18 U.S.C. §1001 and that such willful false statements may jeopardize the validity of the application, any patent issued thereon, or any patent to which this declaration is directed.

Name of Assignee: Dickens-Soeder2000

By: Bruce M. Dickens

Signature: Bruce M. Dickens

**IN THE UNITED STATES PATENT  
AND TRADEMARK OFFICE**

<b>Reissue Application No.</b>	)	
	)	
<b>United States Patent No. 5,806,063</b>	)	<b>Group Art Unit: 2771</b>
	)	
<b>Issued: September 8, 1998</b>	)	<b>Examiner: Wayne Amsbury</b>
	)	
<b>Applicant: Dickens-Soeder2000, LLC</b>	)	<b>Attorney Docket No.: 2039-301</b>
	)	
<b>Attorney Docket No.</b>	)	

**PRELIMINARY AMENDMENT**

Honorable Commissioner of Patents and Trademarks  
Washington, D.C. 20231

Dear Sir:

Applicant in the above referenced Reissue Application to reissue the above  
referenced United States Patent hereby submits this Preliminary Amendment.

Remarks

Claims 1-15 are in United States Patent No. 5,806,063, entitled DATA  
FORMATTING AND SORTING FOR DATES SPANNING THE TURN OF THE  
CENTURY, issued to Dickens on September 8, 1998 on an application filed on October  
3, 1996 ("the patent"), as issued. New claims 16-76 were added to the Reissue  
Application as filed. The patentability of claims 1-15, in light of the Ohms article and the  
Shaughnessy '118 patent has been addressed in the Response of the Patent Owner to the  
Commissioner Initiated Order for Reexamination, filed contemporaneously with the  
patent owner's Reissue Application and this Preliminary Amendment, and incorporated

herein by reference ("the Response"). The patentability of claims 1-15 in light of additional references, including those asserted by Anonymous Submitters in the Reexamination Proceeding to invalidate those claims, and the patentability of the newly submitted claims in light of all references will be addressed in these Remarks. The additional references submitted by the Anonymous Submitters are set forth in Appendix A to this Preliminary Amendment. The patent owner has, in addition, filed an Information Disclosure Statement with the Reissue Application.

References which have been submitted by the Anonymous Submitters, listed in Appendix A, in many cases are not prior art references to the patent. The patent owner, however, in an abundance of caution will discuss these supposed references, but in doing so does not concede that any such supposed reference is actually prior art to the claimed invention as recited in the issued claims or in any of the newly submitted claims. These submitted references will be referred to herein as references, despite the disclaimer of acknowledgement herein that any such material is in fact a prior art reference. Many of these references disclose windowing and in many cases the use of a ten-decade window. None of them discloses the claimed invention, however, nor do any of them, alone or in combination with any others, render the claimed invention obvious. Most, if not all, teach away from the claimed invention.

The obviousness test under §103 must be applied to the subject matter of the claimed invention as a whole. There is no "gist" or "heart" or "core" of the invention evaluated for obviousness purposes. It is necessary to consider all of the subject matter of the claimed invention. Hawes, §7.04 at 7-6, citing, *Loctite Corp. v. Ultraseal Ltd.*, 781

F.2d 861, 228 U.S.P.Q. 90 (Fed. Cir. 1985). Even if the prior art references taken together show all of the elements of the claimed invention, it must still be shown that the claimed invention would have been obvious as a whole and it is improper to analyze the claimed invention by its separate parts, even if each is shown in the art. Hawes, §7.05, at 7-7, citing, *Custom Accessories, Inc. v. Jeffrey-Allan Industries, Inc.*, 807 F. 2d 955, 1 U.S.P.Q.2d 1197 (Fed. Cir. 1986), and *In re Wright*, 848 F.2d 1216, 6 U.S.P.Q.2d 1959 (Fed. Cir. 1988). In the present case, none of the prior art references, nor any combination thereof, discloses all of the elements of the claimed invention, and, therefore, there does not even exist a *prima facie* case for obviousness. See, MPEP § 2143.03, *In re Royka*, 490 F.2d 981, 180 U.S.P.Q. 580 (C.C.P.A 1974), and *In re Wilson*, 424 F.2d 1382, 1385, 165 F.2d 494, 496 (C.C.P.A. 1970). This alone is a basis for finding the claimed invention patentable over these references.

Hindsight must be avoided in combining references in the prior art. It is error to reconstruct the claimed invention from the prior art using the claimed invention as a blueprint. Hawes, §7.05, at 7-8, citing, *Panduit Corp. v. Dennison Manufacturing Co.*, 774 F.2d 1082, 227 U.S.P.Q. 337, 343 (Fed. Cir. 1985), *In re Find*, 837 F.2d 1071, 5 U.S.P.Q.2d 1596 (Fed. Cir. 1988), and *W.L. Gore & Associates, Inc. v. Garlock, Inc.*, 721 F.2d 1540, 220 U.S.P.Q. 303 (Fed. Cir. 1983), *cert. denied*, 469 U.S. 851 (1984). It is improper to reject a claim based upon the mere assertion that one of ordinary skill in the art would know to add a claimed feature to the claimed invention without the citation of a reference that teaches the claimed feature (MPEP §706.02(a)), or at least an affidavit from the Examiner detailing the Examiner's knowledge of the prior art under 37 C.F.R.

§1.107(b). Hawes, §7.08, at 7-19, *citing*, *In re Newell*, 13 U.S.P.Q.2d 1248 (Fed. Cir. 1989), and *In re Kaplan*, 229 U.S.P.Q. 678, 683 (Fed. Cir. 1986).

A reference must be considered as a whole. A prior art reference that describes a product or process similar to the claimed product or process and also a statement that the product or process does not work does not teach the claimed product or process. A prior art reference may be considered to teach away when “ a person of ordinary skill, upon reading the reference, would be discouraged from following the path set out in the reference, or would be led in a direction divergent from the path that was taken by the [inventor].” Hawes, §7.05, at 7-7, *citing*, *In re Gurley*, 27 F.3d 551, 553, 31 U.S.P.Q.2d 1130, 1131 (Fed. Cir. 1994).

Simply because the claimed invention employs a known technique, i.e., windowing, does not, of itself, make the process of the claimed invention obvious. Hawes §7.05, at 7-8.1, *citing*, *In re Brower*, 77 F.3d 422, 37 U.S.P.Q.2d 1663 (Fed. Cir. 1994).

The claimed invention as recited in the original claims 1-15 has been discussed in the Response, and incorporated herein by reference. This discussion applies as well to claims 73 - 76, which are broader than the claims as originally issued. Claims 73 and 74 are identical respectively with claims 1 and 11, except that the recitation specifically of  $M_1M_2$  and  $D_1D_2$  have been omitted. The invention is broad enough to cover an alpha-numeric or other representation, e.g., a Julian representation, of the month and day. Claims 75 and 76 are identical, respectively, to claims 1 and 11 without the express recitation of a ten decade window. In regard to the claimed invention as recited in claims



16-72 the claims are more narrow than those of the original patent and serve to more fully and clearly define the claimed invention over the art.

The art does not teach or suggest "reformatting the symbolic representations of each of the symbolic representations of a date in the database, without the addition of any new data field to the database ... in order to facilitate collectively further processing the reformatted symbolic representations of each of the symbolic representations of each of the dates" as recited in new claims 16 and 26, and the claims that depend from these new claims. The same applies to new claims 31 and 32. The art does not teach or suggest "reformatting the symbolic representations of each of the symbolic representations of a date in the database, without changing any of the symbolic representations of a date in the database during the reformatting step ... in order to facilitate collectively further processing the reformatted symbolic representations of each of the dates" as recited in new claim 33 and the claims depending from that new claim. The art does not teach or suggest "converting each of the symbolic representations of dates stored in the at least one date field of the database to a symbolic representation of each of the respective dates that does not create the ambiguity, by windowing ... without the addition of any new data field to the database for purposes of such windowing and converting; and, running a program collectively on each of the converted symbolic representations of each of the respective dates ..." as recited in new claim 34 and the claims that depend from that new claim. The above comments also apply to new claims 60 and 61. New claims 62-65 include also a feature of "storing the converted symbolic representations separate from the at least one date field of the database" which is not taught or suggested by the art.

The art does not teach or suggest "reformatting the symbolic representations of each symbolic representation of a date in the at least one date field ... and repeating the step of reformatting until each symbolic representation of a date in the at least one date field has been reformatted ... in order to facilitate collectively further processing the reformatted symbolic representations ..." as recited in new claims 66 and 67. The art does not teach or suggest "reformatting ... in order to facilitate further processing of the reformatted symbolic representations of each of the symbolic representations of each of the dates, by running a program on the reformatted symbolic representations of each of the dates" as recited in new claim 68. The art does not teach or suggest "sorting the reformatted symbolic representations of the dates ... and, running a program on the reformatted symbolic representations of each of the dates" as recited in new claim 69. The art does not teach or suggest "converting each of the symbolic representations of dates stored in the at least one date field of the database ... without modifying any of the symbolic representations of dates in the at least one date field ...; and, running a program on the converted symbolic representations of each of the dates ... separately from the date data symbolic representations contained in the at least one date field of the database" as recited in new claim 70. The art does not teach "or suggest storing each of the converted symbolic representations of each of the dates separate from the database; and, running a program on the stored converted symbolic representations of each of the converted symbolic representations ..." as recited in new claim 71. The art does not teach or suggest "reformatting the symbolic representation of each symbolic representation of a date in the database ... prior to collectively further processing

information contained within the database associated with the respective dates” as recited in new claim 72.

Dependent claims 17-25 and 27-30 recite additional features that are patentable over the art for the reasons noted in the Response, incorporated herein by reference. Independent claims 35-59 add recitations to claims 33 and 34, respectively, that also are not taught or suggested by the art in combination with the recitations contained in independent claim 33 or 34, respectively.

The new claims are supported by the original specification and do not add any new matter. The Abstract states that “[d]ates stored in symbolic form in a database are reformatted to permit easy manipulation and sorting of [the] date-related information.” The Abstract also states that “[t]he reformatted date information is particularly useful ... because sorting by date is accomplished using a pure numerical-value sort.” The Specification further states that the “invention relates to the manipulation of information in a database, and in particular, to determination of dates in a useful form.” (Col. 1, lines 7-9) The Specification also notes that “[s]ets of dates spanning the turn of the century ... now stored in many databases ... may be manually converted to a more useable form in the sense that programs may be written to perform conversions, manipulations, and sorting. However, these programs typically require additional data fields for storage, which may be objectionable in some circumstances.” (Col. 1, lines 31-40) The Specification also notes that “there is a need for an improved approach to the representation and utilization of dates in databases, and for converting existing dates in databases to a more useable form.” (Col. 1, lines 41-45) Further notes the Specification, “an approach to the representation and utilization of dates stored symbolically in

databases [includes converting] [e]xisting symbolic representations ... to a more useful form without addition of new data fields, and in a manner that is performed automatically by the computer and requires no user input. The approach of the invention permits numerical sorting of dates.” (Col. 1, lines 51-56) The Specification also notes that “[a] century designator is determined for each date in the database [and] [e]ach date is formatted with the values  $C_1C_2 \dots$ .” (Col. 1, line 66 - Col. 2, line 3) The Specification also notes that “[t]his approach works particularly well if the dates are represented in the format  $C_1C_2 \dots$  [which] may be directly sorted numerically by fast sorting techniques, and thereafter stored back in the database.” (Col. 2, lines 15- 20) In addition, the Specification states “[t]he present invention thus provides an efficient approach to converting and utilizing symbolic date representations in databases, which allows automatic processing of dates ranging from before to after the year 2000. The large number of dates represented in some databases may thereby be readily processed and utilized.” (Col. 2, lines 21-26) The Specification also notes that “[t]his selection process [for the century designators] is performed in a completely automatic fashion by the computer, without human input other than to select the starting date of the 10-decade window.” (Col. 3, lines 36-38) Further, notes the Specification the “symbolic representations of the dates in the database are reformatted with the values  $C_1C_2 \dots$ . In one case that produces particularly advantageous results for many operations, such as chronological date sorting, the date is represented in the form  $C_1C_2 \dots \dots A$  straightforward numerical sort of date fields expressed in this form produces an accurate chronological ordering.” (Col. 3, lines 39-47) The Specification also notes that “[o]nce the symbolic representations of the dates are reformatted according to the procedures set

forth above, the date information may be sorted ... or otherwise manipulated, ... together with the dates, storing the dates and associated information back in the data base, or other processes.” (Col. 3, lines 48-55)

In regard to the references cited by Anonymous Submitter 1:

SAS discloses “YEARCUTOFF+” as a “first year of a 100-year span used as a default by various DATE and DATETIME informats and functions.” It discloses “the relationship between the 100-year span and the two centuries if YEARCUTOFF= 1950.” SQL discloses “RR date format elements ... similar to the YY date format element, but it provides additional flexibility for storing date values in other centuries.” Schinkel discloses using windowing to solve the problem that “[d]ate fields in data entry screens rarely allow you to enter the century ... .” Whitney, discloses “[a]pproaches to fixing [Y2K] problems” including to “[c]hange code to use a date window with YY” but it does not say what to do with the date window, and even cautions against its use “**but ...**”. Straley discloses windowing to “adjust the internal year mechanisms for Clipper ... necessary if there is a string of a date that is to be converted back into the date format [and] for those applications that require string manipulations on dates outside of the currently adjusted century.” Clipper 5.0 discloses “an environment command that determines the interpretation of date strings containing only two year digits.” The July 27 posting describes software of a vendor that is said to employ a technique called “I/O fencing” wherein “[c]onversion code is added to the I/O to expand the date from the old format to the new (the century could be inferred here).” It then suggests that the “an attempt to use it across the board on our large applications, however, would have a high probability of foundering ... .” It also discloses using windowing “when the program

encounters a two-digit year ... ." It then says that this approach has problems when there is "a need to sort data on two-digit years." It also suggests using "MANDATE2, which apparently does windowing to infer a century "for as many date calculations as possible ... ." The July 16 posting discloses a date standard in which "[f]or presentation and input, yy-mm-dd is interpreted" using windowing. The August 7 posting discloses utilizing "sliding dates - leave as two digits & have some logic determine that 60 thru 99 implies 1960 thru 1999 and 00 thru 59 implies 2000 thru 2059." It also notes that "[t]he big problem with this approach is the difficulty of using the data in sorts, etc." The August 8 posting discloses the same thing as the August 7 posting and notes "[t]he programs need to be changed, but the problem still persists. If you sort on dates, look out." The August 3 posting discloses using "procedural logic to determine the century, for example, if year <35 then century =20 , else century = 19." It also notes that "[p]robably the most desirable path is to convert all two digit years to four digit years. Interface programs will be required for systems that have not been changed."

None of these disclose or suggest the claimed invention as recited in claims 1-15 as issued originally in the patent. To the extent they disclose windowing they utilize it in the manner of the '118 patent, on-call to process at most two date data entries at a time, and note the attendant problems, e.g., in doing sorting, or they utilize windowing in the manner of Roth et al., United States Patent No. 5,878,422, entitled SYSTEM FOR VIRTUALLY CONVERTING DATA IN A FIELD BETWEEN FIRST AND SECOND FORMAT BY USING HOOK ROUTINES, issued on March 2, 1999 on a application filed on April 9, 1997, which was a continuation-in-part of an application dated April 9, 1996, and assigned to Viasoft, Inc. Roth et al. has an effective date later than the date of

invention of the claimed invention, as demonstrated in the prosecution of the patent and is, therefore, not a prior art reference. Roth et al., however, does disclose a system which “assumes control each time a converted program reads or writes a record of a data file that has not yet been converted [and converts] records during I/O operations [enabling] a converted program to access an unconverted data file as if it were converted.” (Col. 2, lines 59-64) The disclosed embodiments employ a form of on-call windowing in a technique of I/O fencing by “intercept[ing] input/output (I/O) read requests from the program [and] intercept[ing] input/output (I/O) write requests from the program, and utilizing the conversion routine ... .” (Col. 3, lines 56-61) One form of conversion routine is windowing. (Col. 11, lines 20- 26).

Anonymous Submitter 3 filed a Request for Reexamination after the Commissioner had already ordered Reexamination. The patent owner herein addresses the art cited by Anonymous Submitter 3 as if submitted in the Reexamination proceeding initiated by the Commissioner. The citations of Anonymous Submitter 3 include Ohms and ‘118, discussed in the Response. The additional citations of Anonymous Submitter 3 also do not teach or suggest the claimed invention. Browe teaches “[d]ate calculations for FOCUS files.” He notes that “[a]ll available dates are in the form MMDDYYYY, using a 4 digit year. This is done to avoid additional program logic to calculate dates between the 20<sup>th</sup> and 21<sup>st</sup> century.” He also discloses a process to “change a date to other than the current date [in order to] ‘recreate’ accounting reports for previous accounting periods or preview future reports without changing program logic.” In doing this Browe teaches that “[t]he reference date is typed in MMDDYY format and must refer to a date from 01/01/1950 to 12/31-2049 for this focexec to work correctly.” He also notes that in

setting this "REFDATE" "[i]f the year is between 50 and 99, the date is assumed to be in the 20<sup>th</sup> century (1900-1999) else the 21<sup>st</sup> century (2000-2099)." Japan 05-027947 teaches a "2000 AD correspondence utility module [that is] activated when specified by program (7) or a utility and is positioned as pre-processing for processing that handles the year." (Translation, at 4). The Millenium Journal teaches that "[t]he following approaches to dealing with dates are not recommended." One such approach involves one form of "[l]logic-based century determination," i.e. windowing, which "is only recommended as a temporary measure." After describing windowing the Journal notes that "[s]orts require an exit, and different date fields may require a different frame. Again, every place where dates are used in calculations, comparisons or displays, it is necessary to add additional logic." Milam teaches a system, like Ohms that stores and manipulates dates and times as a single binary number which represents the number of seconds that have elapsed since January 1, 1970 and teaches ways to convert this number into representations, including "building date strings." (at 72) He teaches "tp\_date takes a date string plus a format indicator and calculates a date (date\_t) value." "If the century in the date string is omitted, to\_date uses [windowing] to determine the century." (at 72-73) Lysgaard "deals with the problems [omission of century dates] causes in EDP systems as we approach the year 2000." He teaches that "[i]f information as to the valid time interval is added to the programme - maybe just the start for the 100 year interval - the programmes will be able to handle all time calculation correctly." (at 515) He also notes that "[a]lready today, most date manipulations take place via standard routines and ... as we approach year 2000, the standard routine library should probably be extended ... ." He also notes that "[w]hen one chooses to change only the programmes it is



necessary to discover which operations in the programmes will go wrong in year 2000. Such an analysis is also a fine basis for judging which standard routines that should be offered ... .” One of the areas on the “problem list” includes “[s]orting of 2 digit years with usual utility programmes will result in erroneous chronological number order. Correct number order is achieved as follows:

definition of sorting sequence for first digit in year (decade), for example 6789012345 for time interval 1960-2059. This method can only be used if the interval is at the start of a decade.

temporary addition of auxiliary fields stating the century and included in the sorting criteria.

a sorting routine which can sort with transposed zero point. If the transposition for a 2 digit number area is set at ‘60’, the order of the sorting should be 60-99, 00-59.

Lysgaard also notes that “[a] special problem presents itself if one has a 6 digit date included in the primary key of a file organized index sequentially.” (at 516-17).

Shaw teaches windowing, but not specifically how to employ windowing, i.e., he does not teach the claimed invention. SAA AD/Cycle Language Environment, Programming Guide teaches a system, like Ohms, in which the dates are stored and manipulated in Lilian format (at xvi) and a number of “callable services” (at 31) which include CEEDAYS which uses windowing, like Ohms, to convert date strings into Lilian even if the century designator is not present. (at 84). SAS Language: Reference teaches “date and time informats that instruct the SAS System to read data values into variables that represent dates, times, and datetimes.” (at 62) Similar to Ohms, SAS stores and

manipulates dates as a number representing the number of days since January 1, 1960. (at 85, 129)

None of the citations of Anonymous Submitter 3 teach or suggest the claimed invention as recited in claims 1-15 as originally issued in the patent. Neither do any of them teach or suggest the more narrowly defined claimed invention as recited in new claims 16-72 or the broadened claims 73-76.

Other prior art patents than those discussed above, under §102 (b), or under §102 (e), with effective dates before the date of the invention of the claimed invention, also do not teach or suggest the claimed invention. United States Patent No. 5,471,631, entitled USING TIME STAMPS TO CORRELATE DATA PROCESSING EVENT TIMES IN CONNECTED DATA PROCESSING UNITS, issued to Beardsley et al. on November 28, 1995 on an application dated October 19, 1992 teaches "a host processor(s) [that] is loosely-coupled by a plurality of data channels to a peripheral subsystem(s). The host processor(s) has a time of day clock. The peripheral subsystem(s) has a cluster(s) that performs peripheral controller functions. Each cluster has first and second clocks that respectively generate a log entry (logging) reference time and a subsystem time. The first and second clock times result in subsystem time stamps in a log that are not related to the time of day clock. For time correlating the time of day clock with the first and second clocks, a host time stamp is generated from the time of day clock." (Abstract) United States Patent No. 5,526,515, entitled HARDWARE-DRIVEN CLOCK WITH OBJECT-ORIENTED INTERFACE, issued to Ross et al. on June 11, 1996, on an application dated June 30, 1994 teaches using "a floating point number ... for the total tick count to represent time so that both a large range and a high resolution can be accommodated," as

shown in Table I. (Col. 3, lines 29-31) United States Patent No. 5,680, 622, entitled SYSTEM AND METHODS FOR QUICKLY DETECTING SHAREABILITY OF SYMBOL AND TYPE INFORMATION IN HEADER FILES, issued to Even on October 21, 1997 on an application filed on June 30, 1994 discloses a computer system which "eliminat[es] the storage of redundant data which is routinely present in parsed information." (Col. 3, lines 25-26) This is done by storing information in a "parse tree." (Col. 3, line 37) United States Patent No. 5,513,351, entitled PROTECTING A SYSTEM DURING SYSTEM MAINTENANCE BY USEAGE OF TEMPORARY FILENAMES IN AN ALIAS TABLE, issued to Grantz on April 30, 1996 on an application filed on July 28, 1994, discloses a "method of maintaining functional compatibility after performing system maintenance on a computer during system operation. The method ... determin[es] if a directory holding a set of files to be maintained has adequate space for a set of new files [and] in response to there being adequate space for the new set of files, places the new set of files in the directory. ... The new names of the new set of files are then placed in an alias table for access by any other files open on the computer during maintenance." (Abstract) United States Patent No. 5,513,306, entitled TEMPORAL EVENT VIEWING AND EDITING SYSTEM, issued to Mills et al. on April 30, 1996 on an application filed on June 7, 1995, which was a continuation of an application filed on April 11, 1994, which was a continuation of an application filed on August 9, 1990, discloses "a system for displaying a sequential stream of information on a single display." (Abstract)

A number of other patents having effective dates prior to the filing date of the patent, but not before the date of invention, address the Y2K problem in ways that also do

not teach or suggest the claimed invention. United States Patent No. 5,838,979, entitled PROCESS AND TOOL FOR SCALABLE AUTOMATED DATA FIELD REPLACEMENT, issued to Hart et al. on November 17, 1998 on an application filed on October 31, 1995, discloses "global modification of complex collections of interacting computer programs" among other things, for purposes of addressing the problem that "once these programs [which assume "that the two least significant digits of a year were adequate {sic} indicate the year"] begin to address years 2000 and beyond, there will be errors." (Col. 1, lines 44-51) Hart et al. teaches using "a global attribute database ... with the attributes of all variables which are subroutine parameters or arguments." Col. 32, lines 34-36) Also Hart et al. teaches "translation of the old program into an intermediate language." (Col. 2, lines 42-43) Hart et al. also teaches "expanding the year field to four digits," and "the addition of the logic for a century indicator where only a two-digit year representation is used and the user chooses to add the century indicator as a solution to the year representation problem." (Col. 10, lines 48-51) United States Patent No. 5,600,836, entitled SYSTEM AND METHOD FOR PROCESSING DATE-DEPENDENT INFORMATION WHICH SPANS ONE OR TWO CENTURIES, issued to Alter on February 4, 1997 on an application filed on November 14, 1995 discloses a "system that processes date-dependant information in which dates are specified in a two-digit format and are in at most two centuries. The system ... includes a number of local data files having dates represented in local time, a number of zone data files having dates represented in zone time, and at least one time change interface between local date files and zone date files that converts the dates represented in local time to dates represented in zone time and converts the dates represented in zone time to dates represented in local

time.” (Col. 2, lines 8-18) The local time dates are reduced by an amount in forming the zone time dates such that all zone time dates are within a single century and processing is done on them as if they were within a single century. (Col. 5, line 41-67) United States Patent No. 5,761,668 entitled METHOD AND APPARATUS FOR CONVERTING COMPUTER SOFTWARE AND DATABASES FOR THE YEAR 2000, issued to Adamchick on June 2, 1998 on an application dated October 29, 1996, claiming priority of a provisional application dated March 8, 1996, teaches a “method of modifying existing computer software and databases to utilize a date representation which accommodates dates subsequent to December 1, 1999 but preserves the desirable properties of utilizing no more than six numeric characters [with the] date ... represented as CYYDDD ... .” (Abstract) Adamchick teaches that windowing is a way to convert dates for the Y2K solution, but that Adamchick’s method is different. (Col. 3, lines 53-67) United States Patent No. 5,878,422, entitled SYSTEM FOR VIRTUALLY CONVERTING DATA IN A FIELD BETWEEN FIRST AND SECOND FORMAT BY USING HOOK ROUTINES, issued to Roth et al. on March 2, 1999, on an application filed on April 9, 1997, which was a continuation-in-part of an application filed on April 9, 1996, is discussed above. United States Patent No. 5,719,826, entitled CALENDARING SYSTEM, issued to Lips on February 17, 1998 on an application filed on May 1, 1996 teaches “[a] calendaring system which provides accurate calendaring of all dates between Jan. 01, 0001 and Dec. 31, 9999 by implementation of an extended set of leap-year rules.” (Abstract) The system stores and manipulates days in integer absolute days starting with January 1, 0001. (Col. 4, lines 38-50) The system “maintains consistency between two basic types of functions ... comput[ing] the number of days

between to dates [and] calculat[ing] a date based upon a starting date and a numerical increment.” (Col. 4, line 63 - Col. 5, line 2) A “parameter CENTURY-BREAK allows the user to tell the system how to assign a value to the century when the supplied date does not explicitly contain a century [using windowing].” (Col. 7, lines 1-12) United States Patent No. 5,644,762, entitled METHOD AND APPARATUS FOR RECORDING AND READING DATE DATA HAVING COEXISTING FORMATS, issued to Soeder on July 1, 1997 on an application filed on May 14, 1996, discloses a system for “using at least one of these unused ranges to integrate at least two formats for encoding the year data . . . . For new date data, i.e., after Dec. 31, 1999, the integer value of the year number is represented in binary format, and the binary representation is written in the two bytes already allocated for the year.” (Col. 2, lines 8-14) United States Patent No. 5,758,336, entitled DATE FORMAT AND DATE CONVERSION PROCEDURE USING A PACKED BINARY FORMAT, issued to Brady on May 26, 1998 on an application dated May 30, 1996, teaches “[a] binary date format, . . . ‘packed binary’, . . . compris[ing] a 2-byte data field. The first two bits of the data field are utilized as identifiers to enable the packed binary format to be identified. The remaining bits are employed as a binary value year indication. A date conversion procedure employs the packed binary format and is called when a program reaches a date field that is to be processed. The date conversion procedure initially examines the date field to determine whether its format is zone decimal, packed decimal or packed binary. If the date field data format is either zone decimal or packed decimal, the date values are converted to packed decimal and the required arithmetic procedure is performed, using the packed binary format date value. Once the calculation is complete, the procedure determines whether a “year 2000” switch

is set--indicating that all results are to be returned to the program in packed binary format. If the 2000 switch is not set, the date conversion procedure reconverts the packed binary year format back to the originally received format and returns the result to the running program.” (Abstract) United States Patent No. 5,970,247, entitled METHODS FOR ENCODING DECODING AND PROCESSING SIX CHARACTER DATE DESIGNATIONS FOR THE YEAR 2000 AND BEYOND, issued to Wolf on October 19, 1999 on an application filed on October 7, 1997, claiming priority from a number of provisional applications going back to June 17, 1996, discloses “methods for encoding (and decoding) calendar dates in multiple centuries, as sequences of six characters.” (Col. 2, lines 64-67) Wolf teaches the use of “non-numeric characters, or of four-digit numbers greater than 1231, to represent the month and day in dates in and after a specified year, e.g., the year 2000.” United States Patent No. 5,808,889, entitled SYSTEM AND METHOD FOR IDENTIFYING AND CORRECTING COMPUTER OPERATIONS INVOLVING TWO DIGIT YEAR DATES, issued to Burgess on September 15, 1998 on an application dated June 28, 1996, teaches a “system and method for identifying and correcting computer operations involving two digit year dates. The system includes: a computer and a routine for searching for a subtraction, comparison and sort operations on the computer involving a plurality of two digit quantities representing year dates. The subtraction operation is executed to obtain a difference value between the two quantities. A negative difference value is verified and a plurality of two digit numbers whose sum is 100 is added to the negative difference value so that the subtraction operation yields a correct result for two digit quantities representing years greater than or equal to the year 2000. A similar routine is added to existing computer code involving compare and/or

sort operations.” (Abstract) United States Patent No. 5,740,442, entitled METHOD AND APPARATUS FOR IDENTIFYING AND CORRECTING DATE CALCULATION ERRORS CAUSED BY TRUNCATED YEAR VALUES, issued to Cox, et al. on April 14, 1998 on an application filed on August 27, 1996 teaches a “standardized test environment [that] assists identifying problematic instances of instructions in computer programs that produce erroneous results caused by dates having years expressed in a truncated form. ... In one embodiment of a production environment, a computer system intercepts execution of a computer program at specified locations and takes corrective action according to information obtained from a table of control information.” (Abstract)

United States Patent No. 5,794,048, entitled METHOD FOR CLASSIFICATION OF YEAR-RELATED DATA FIELDS IN A PROGRAM, issued to Brady on August 11, 1998, on an application filed on August 29, 1996, teaches “enabl[ing] a computer to examine a software application composed of one or more programs, which includes operand labels and logical or computational operators, and to identify operand fields which include a year value. ... Thereafter, the method determines, for operator entries located in the operator table, and from operands associated therewith and an operand association table, whether the classification for each respective operand in the operand table should or should not be altered.” (Col. 2, lines 8-28) Brady also teaches that this procedure “enables identification of year fields with high probability, and thereafter enables each of these fields to be automatically accessed and to have their year designation altered in such a manner as to enable a removal of the ambiguity which arises from the onset of the year 2000 and forward. A preferred method for revising the year format for enabling the year conversion procedure is described in U.S. patent application



Serial No. 08/657,657 ... entitled 'Date Format and Date Conversion Procedure' to Brady." (Col. 9, line 63 - Col. 10, line 5) United States Patent No. 5,956,510, entitled APARATUS AND METHOD FOR REVISING COMPUTER PROGRAM CODE, issued to Nicholas on September 21, 1999 on an application filed on September 10, 1996 teaches an approach to the Y2K problem that is asserted to be in lieu of an "attempt to expand a two-digit field to four digits," (Col. 1, lines 37-38) or using "logic that assumes that the century has changed if the year value falls into a specified range." (Col. 1, lines 51-53) Nicholas teaches an "apparatus and methods for revising a field in computer program code [which] includes the steps of analyzing program code lines for the presence of the field therein and, if present, revising the field from the first value range to a second value range." (Col. 2, lines 13-17) Nicholas further teaches "inserting an extended field representative of a base value into the program code, with the field being encoded as an offset value relative to the extended field." (Col. 2, lines 21-21) United States Patent No. 5,668,989, entitled TWO-DIGIT HYBRID RADIX YEAR NUMBERS FOR YEAR 2000 AND BEYOND, issued to Mao on September 16, 1997 on an application filed on September 18, 1996, discloses the use of "numbers ... represented with 2 digit hybrid radix numbers, in which the higher digit is input, generated, processed, and output as hexadecimal, but displayed in a decimal-like way with font patterns 0-9 and '0-'5, while the lower digit is treated as ordinary decimal." (Col. 2, lines 37-41) In this manner "no changes to the existing application source code other than a re-compilation is needed, and compatibility with existing data files [sic files] and databases is automatically maintained. All data files need no change, and all the databases will continue to work

with a re-compiled DBMS while no change is needed for the database schema and contents.” (Col. 2, lines 44-50)

Important discussions of the Y2K problem around the time of the invention of the claimed invention also teach away from the claimed invention. These include the IBM publications, *The Year 2000 and 2-Digit Dates: a Guide for Planning and Implementation*, IBM, First Edition, October 1995 (“the Guide, First Ed.”), and *The Year 2000 and 2-Digit Dates: a Guide for Planning and Implementation*, IBM, Third Edition, May 1996 (“the Guide, Third Ed.”), each of which was of record and sworn behind in the initial prosecution of the patent, and J. Murray and M. Murray, *The Year 2000 Computing Crisis*, McGraw-Hill, 1996, Chapter 11, The Conversion Plan, (“Murray”), which was submitted anonymously in the Reexamination proceeding by Anonymous Submitter 2, and asserted by Anonymous Submitter 2 to have been published in March of 1996.

The Guide First Ed. does not teach or suggest the claimed invention. It notes in Chapter 5, at 5-1, which contains “a number of techniques ... to correct improper date notation and use” that the number one choice is “[c]onversion to [f]ull 4-[d]igit-[y]ear [f]ormat, [which] requires changes to both the data and the programs by converting all references and/or uses of 2-digit year format (YY) to 4-digit-year format (YYYY).” (*Id.* at 5-1)

The Guide First Ed. then suggests using a form of windowing (“Fixed Windowing Technique”) such that “[p]rogram logic determines the century based on the following data checking. If the 2-digit year representation of a specific year is  $xy$  then if:

$xy \geq 60$ , then it is a 20<sup>th</sup> century date (19 $xy$ )

otherwise (that is,  $xy \leq 59$ ), it is a 21<sup>st</sup> century date (20 $xy$ ).”

This suggestion is to change dates in an on-call routine in the manner of the Shaughnessy '118 patent discussed in the Response. The Guide, First Ed. notes that in using this technique the user should "[e]xpect a performance impact in direct proportion to the quantity of date processing the particular application handles due to the overhead of 2- to 4- digit year conversion." The Guide First Ed. also notes that "[a]ll programs that use the fixed window technique may need to be manually updated on a yearly basis depending on how your date routine is packaged." In addition the Guide First Ed. notes "[a]ll programs that accept output from the fixed window technique must use the same assumptions (current date, past and future windows)."

Finally the Guide First Ed. notes that:

[r]etaining a 2-digit year representation does not provide collating sequence support. Nor does the use of a fixed window technique provide indexing sequence support when 2 digit years are used as index keys in indexed files. You need to provide additional processing to obtain correct collating and indexing sequence output.

The Guide, First Ed. also suggests the use of a "Sliding Window Technique." Except for the "manually updated" requirement, the above comments are made with respect to the sliding windowing technique as well.

The Guide, First Ed. does not teach doing sorting utilizing the windowing technique disclosed in the Guide, First Ed. In either the Guide Second Edition or the Guide Third Ed., as appears in the Guide, Third Ed., there is added the following sentence to the description of the sliding windowing technique:

[f]or an example of how DFSORT/MVS is implementing a sliding window to sort, merge, and transform 2-digit year data to 4-digit year data, refer to "DFSORT" on page 7-23.

The section "DFSORT" as shown in the Guide, Third Ed., at 7-23, and the Guide, Ninth Edition, at 8-25, does not appear in the Guide, First Ed. The DFSORT discussion also indicates that the Guide, Third Ed. does not teach or suggest the claimed invention:

In Murray, the authors suggest a "conversion method" which utilizes windowing, but not in the manner of the claimed invention. It teaches the total conversion of all database date entries to four character date data, along with all of the programs. Indeed, it teaches that to "create a program environment whose logic is internally eight-digit-date-oriented while all of its output is six-digits dates[ i.e., to] patch programs without having to update databases" is **not** the appropriate approach. (*Id.* at 204) "Conversion to an eight-digit-date standard both within programs and within databases staves off ultimate disaster and preserves the investment demanded by the current [Y2K] crisis."(*Id.*)

A number of patents which are clearly not part of the prior art have been issued by the United States Patent and Trademark Office which show various approaches to solving the Y2K problem. Many of these disclose or imply various uses of windowing in the contexts of the differing solutions taught and suggested in these patents. These patents include United States Patent No. 5,806,067, entitled METHOD AND APPARATUS FOR DATA PROCESSING WITH CONVERSION OF AMBIGUOUS DATA STRUCTURES, issued to Connor on September 8, 1998 on an application filed on October 21, 1996; United States Patent No. 5,845,286, entitled DATE VALUE

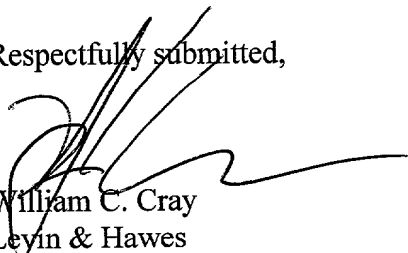
REDUCTION SYSTEM, issued to Colizza on December 1, 1998, on an application dated December 24, 1996; United States Patent No. 5,978,809, entitled, METHOD OF SOLVING MILLENNIUM PROBLEMS OF SOME APPLICATION PROGRAMS, issued to Bemer on November 2, 1999 on an application filed on January 27, 1997; United States Patent No. 5,758,346, entitled CONVERTING REPRESENTATIONS OF YEAR, issued to Baird on May 26, 1998 on an application filed on January 29, 1997; United States Patent No. 5,809,500, entitled SYSTEM FOR CONVERTING PROGRAMS AND DATABASES TO CORRECT YEAR 2000 PROCESSING ERRORS, issued to Nolan on September 15, 1998, on an application filed on June 20, 1997, claiming priority from a provisional application filed on February 26, 1997; United States Patent No. 5,915,116, entitled TIME VALUE MANIPULATION, issued to Hochman et al. on June 22, 1999, on an application filed on March 7, 1997; United States Patent No. 6,009,540, entitled LOGIC MODULE FOR IMPLEMENTING SYSTEM CHANGES ON PC ARCHITECTURE COMPUTERS, issued to Craft et al. on December 28, 1999, on an application filed on April 8, 1997; United States Patent No. 5,903,895, entitled METHOD FOR REFORMATION CONVENTIONAL THREE FIELD DATE FORMATS TO PRODUCE A CENTURY ACCUMULATED DATE, issued to Hoffman et al. on May 11, 1999, on an application dated April 29, 1997; United States Patent No. 5,852,824, entitled APPARATUS AND METHOD FOR PROCESSING YEAR-DATE DATA IN COMPUTER SYSTEMS, issued to Brown on December 22, 1998 on an application filed on May 22, 1997; United States Patent No. 5,797,117, entitled MONTH FIELD DIVISION MULTIPLEXING SOLUTION FOR YEAR 2000 COMPUTER DATE PROBLEM, issued to Gregovich on August 18, 1998

on an application filed on May 24, 1997; United States Patent No. 5,802,354, issued to Kubala et al. on September 1, 1998, on an application filed on June 9, 1997, entitled METHOD AND APPARATUS FOR SYNCHRONIZING SELECTED LOGICAL PARTITIONS OF PARTITIONED INFORMATION HANDLING SYSTEM TO A TEST DATASOURCE; United States Patent No. 5,911,142, entitled SYSTEM AND METHOD FOR BRIDGING COMPLIANT AND NON-COMPLIANT FILES, issued to Smith et al. on June 8, 1999, on an application filed on August 5, 1997, claiming priority of a provisional application filed on July 1, 1997; United States Patent No. 6,002,873, entitled, COMPILER-ASSISTED OR INTERPRETER ASSISTED WINDOWING SOLUTION TO THE YEAR 2000 PROBLEM FOR COMPUTER PROGRAMS, issued to Carter et al. on September 14, 1999 on an application filed on November 14, 1997, which was a continuation in part of an application filed on July 23, 1997; United States Patent No. 5,765,145, entitled METHOD FOR PERFORMING DATE OPERATIONS ON DATE FIELDS SPANNING SEVERAL CENTURIES, issued to Masiello on June 9, 1998 on an application filed on July 28, 1997; United States Patent No. 5,926,814, entitled SYSTEM AND METHOD FOR PROCESSING A NEW CALENDAR SYSTEM, issued to Friedman on July 10, 1999, on an application filed on September 22, 1997; and United States Patent No. 6,003,028, entitled IMPLEMENTING EXTENDED NUMERIC RANGE WITHIN A TWO DIGIT SOFTWARE REPRESENTATION, issued to Koenig on December 14, 1999 on an application filed on January 13, 1998 which claimed priority from a provisional application filed on December 16, 1997.

For the above stated reasons, the applicant submits that the claimed invention as recited in claims 1-15 of the patent, as issued, and in the new claims 16-76 added in the

Reissue Application are patentable and the claims of the patent should be reissued and the new claims 16-76 allowed and added to the reissued patent.

Respectfully submitted,



William C. Cray  
Levin & Hawes  
Attorneys for Dickens-Soeder2000  
February 22, 2000

# United States Patent [19]

Dickens

---

[54] DATE FORMATTING AND SORTING FOR  
DATES SPANNING THE TURN OF THE  
CENTURY

[75] Inventor: Bruce Dickens, Irvine, Calif.

[73] Assignee: McDonnell Douglas Corporation,  
Long Beach, Calif.

[21] Appl. No.: 725,574

[22] Filed: Oct. 3, 1996

[51] Int. CL<sup>6</sup> ..... G06F 17/30

[52] U.S. CL. .... 707/6; 707/102; 707/7;  
707/200

[58] Field of Search ..... 707/6, 102, 7,  
707/200

[56] References Cited

## U.S. PATENT DOCUMENTS

4,573,127	2/1986	Kotff .....	364/493
5,630,118	5/1997	Shanghnessy .....	707/1
5,644,762	7/1997	Soeder .....	707/6
5,668,989	9/1997	Mao .....	707/101



# DATE FORMATTING AND SORTING FOR DATES SPANNING THE TURN OF THE CENTURY

## BACKGROUND OF THE INVENTION

This invention relates to the manipulation of information in a database, and, in particular, to the determination of dates in a useful form.

Dates are stored as symbolic representations in computer databases in varying formats. For example, a date may be represented in the numerical representation MM/DD/YY, where MM is a two-digit month designator, DD is a two-digit day designator, and YY is a two-digit year designator (the last two digits of the year). Thus, Dec. 15, 1993 is designated as 12/15/93. A date may also be represented in an alphanumeric form MMM/DD/YY, where MMM is an alphabetic month designator (e.g., DEC for December), and DD and YY are the same as in the numerical form. Dec. 15, 1993 is represented in this format as DEC/15/93.

Such approaches for the representation of dates have worked well since the advent of computer databases, which has occurred in the twentieth century. Dates may be sorted in chronological order using the numerical representations. However, with the turn of the century at Jan. 1, 2000, the representation and utilization of dates becomes more complex. Using the numerical form above, Dec. 15, 2000 is represented as 12/15/00. If a numerical sort is performed on 12/15/93 and 12/15/00, the later date 12/15/00 sorts as the first-occurring date, an incorrect result.

Sets of dates spanning the turn of the century and associated with past, current, and future activities are now stored in many databases. When stored in the conventional formats discussed above, those dates will not readily be used and numerically sorted in chronological order. They may be manually converted to a more usable form in the sense that programs may be written to perform conversions, manipulations, and sorting. However, these programs typically require additional data fields for storage, which may be objectionable in some circumstances.

There is a need for an improved approach to the representation and utilization of dates in databases, and for converting the existing dates in databases to a more usable form. The present invention fulfills this need, and further provides related advantages.

## SUMMARY OF THE INVENTION

The present invention provides an approach to the representation and utilization of dates stored symbolically in databases. Existing symbolic date representations are converted to a more useful form of symbolic date representations without the addition of new data fields, and in a manner that is performed automatically by the computer and requires no user input. The approach of the invention permits direct numerical sorting of dates.

In accordance with the invention, a method of processing dates stored in a database comprises the steps of providing a database with dates stored therein according to a format wherein  $M_1M_2$  is the numerical month designator,  $D_1D_2$  is the numerical day designator, and  $Y_1Y_2$  is the numerical year designator, all of the dates falling within a 10-decade period of time. A 10-decade window with a  $Y_A Y_B$  value for the first year of the ten-decade window is selected,  $Y_A Y_B$  being no later than the earliest  $Y_1 Y_2$  year designator in the database. A century designator  $C_1 C_2$  is determined for each date in the database,  $C_1 C_2$  having a first value if  $Y_1 Y_2$  is less

than  $Y_A Y_B$  and having a second value if  $Y_1 Y_2$  is equal to or greater than  $Y_A Y_B$ . Each date in the database is formatted with the values  $C_1 C_2$ ,  $Y_1 Y_2$ ,  $M_1 M_2$ , and  $D_1 D_2$ .

In the case of most practical interest, the 10-decade period of time spans the year 2000 and begins with a year in which the second digit ( $Y_B$  in  $Y_A Y_B$ ) is 0 (zero). For any 10-decade period including the year 2000, if the decade designator  $Y_1$  of the date in the database is numerically less than the decade designator  $Y_A$  of the first decade of the 10-decade period of time, the century designator  $C_1 C_2$  is "20". If  $Y_1$  is equal to or greater than  $Y_A$ ,  $C_1 C_2$  is "19". Dates in databases spanning more than 10 decades are not handled by this approach, but it is not expected that this limitation will be significant for most commercial and industrial databases.

This approach works particularly well if the dates are represented in the format  $C_1 C_2 Y_1 Y_2 M_1 M_2 D_1 D_2$ . The date Dec. 15, 2000 is represented in this format as 20001215, for example. Dates represented in this format may be directly sorted numerically by fast sorting techniques, and thereafter stored back in the database.

The present invention thus provides an efficient approach to converting and utilizing symbolic date representations in databases, which allows automatic processing of dates ranging from before to after the year 2000. The large number of dates represented in some databases may thereby be readily processed and utilized. Other features and advantages of the present invention will be apparent from the following more detailed description of the preferred embodiment, taken in conjunction with the accompanying drawings, which illustrate, by way of example, the principles of the invention. The scope of the invention is not, however, limited to this preferred embodiment.

#### BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a schematic representation of a computer database with date information therein; and

FIG. 2 is a block flow diagram of a preferred approach for practicing the approach of the invention.

#### DETAILED DESCRIPTION OF THE INVENTION

FIG. 1 schematically depicts a computer 20 having a read-only or random-access memory 22, a mass-storage device 23, and a central processing unit 24 therein. Stored in the memory 22 or on the mass-storage device 23 is a database 26. The database includes information in the form of symbolic representations of dates and associated information such as events occurring on the respective dates. In a conventional approach, the dates are stored in a format such as  $M_1 M_2 / D_1 D_2 / Y_1 Y_2$  format. M indicates month information, D day information, and Y year information, with the subscript 1 or 2 indicating the first or second digit of the designator, respectively. Dec. 15, 1993 is stored as 12/15/93 or 12-15-93, and Dec. 15, 2000 is stored as 12/15/00 or 12-15-00, for example. If a numerical sort is performed on these dates, 12/15/00 will sort chronologically prior to 12/15/93.

FIG. 2 illustrates the approach of the invention. The computer database 26 is provided, numeral 30, having symbolic representations of dates stored therein. In some cases, the dates will be represented as discussed in the preceding paragraph. In other cases, an alphanumeric designator is used. In that approach, each date is stored as  $M_c M_m M_d / D_1 D_2 / Y_1 Y_2$  format, where  $M_c M_m M_d$  is an alphabetical symbol such as JAN for January, FEB for February,

etc. In that case, the month designator  $M_1M_2$  is first converted to the numerical form  $M_1M_2$  by converting JAN to "01", FEB to "02", etc.

A 10-decade window is selected, numeral 32. That is, it is necessary that all dates in the database will be within some period of 10 decades, or 100 years. This limitation poses little problem for most industrial and commercial databases. The window may be arbitrarily selected. For example, the decade could begin with the 1950's and end with the 2040's, or it could begin with the 1980's and end with the 2070's. The 10-decade window will normally include some decades from the prior century and some from the new century.

The first year of the 10-decade window is represented by  $Y_1Y_2$ . In a commonly utilized application,  $Y_2$  is 0 (zero), although the invention is not limited to this case. That is, the 1950's first decade would be represented by  $Y_1Y_2$  of "50", and the 1980's first decade would be represented by  $Y_1Y_2$  of "80". For this case, a century designator  $C_1C_2$  for a date is determined, numeral 34, by comparing the value of  $Y_1$ , the first digit of the year designator for the date, with  $Y_A$ , the first digit of the first decade of the 10-decade window.  $C_1C_2$  is assigned a first value if  $Y_1$  is less than  $Y_A$  and a second value if  $Y_1$  is equal to or greater than  $Y_A$ .

In the case of most interest, the 10-decade window includes decades earlier than the year 2000 and decades later than the year 2000, and  $Y_2$  is zero.  $C_1C_2$  is assigned "20" if  $Y_1$  is less than  $Y_A$  and is assigned "19" if  $Y_1$  is equal to or greater than  $Y_A$ . In that case and for example, if  $Y_A$  is 5, meaning that the decade beginning in 1950 was selected as the first decade of the 10-decade window, and if  $Y_1Y_2$  is "43", the century designator  $C_1C_2$  is "20", indicating that the year in question in the database is 2043. On the other hand, if  $Y_1Y_2$  is "63", the century designator  $C_1C_2$  is "19", indicating that the year in question in the database is 1963. This selection process is performed in a completely automated fashion by the computer, without human input other than to select the starting date of the 10-decade window.

The symbolic representations of the dates in the database are reformatted with the values  $C_1C_2$ ,  $Y_1Y_2$ ,  $M_1M_2$ , and  $D_1D_2$ , numeral 36 of FIG. 2. In one case that produces particularly advantageous results for many operations, such as chronological date sorting, the date is represented in the form  $C_1C_2Y_1Y_2M_1M_2D_1D_2$ . For example, the date 12/15/93 (Dec. 15, 1993) is represented as 19931215 and the date 12/15/00 (Dec. 15, 2000) as 20001215. A straightforward numerical sort of date data fields expressed in this form produces an accurate chronological ordering.

Once the symbolic representations of the dates are reformatted according to the procedures set forth above, the date information may be sorted, numeral 38, or otherwise manipulated, numeral 40, together with the entries associated with the dates. Such manipulation may include handling of data associated with the dates, storing the dates and associated information back in the data base, or other processes.

The approach of the invention has been implemented in a computer program, a copy of which is attached as Exhibit A. This program converts dates both before and after the year 2000.

The present invention provides an effective technique for reformatting symbolic representations of date information that is rapid and automated, and yields new symbolic representations of date information that are particularly amenable to further processing. Although a particular embodiment of the invention has been described in detail for purposes of illustration, various modifications and enhance-

ments may be made without departing from the spirit and scope of the invention. Accordingly, the invention is not to be limited except as by the appended claims.

What is claimed is:

- 5 1. A method of processing symbolic representations of dates stored in a database, comprising the steps of  
 providing a database with symbolic representations of dates stored therein according to a format wherein  
 10  $M_1M_2$  is the numerical month designator,  $D_1D_2$  is the numerical day designator, and  $Y_1Y_2$  is the numerical year designator, all of the symbolic representations of dates falling within a 10-decade period of time;  
 selecting a 10-decade window with a  $Y_A Y_B$  value for the first decade of the window,  $Y_A Y_B$  being no later than  
 15 the earliest  $Y_1Y_2$  year designator in the database;  
 determining a century designator  $C_1C_2$  for each symbolic representation of a date in the database,  $C_1C_2$  having a first value if  $Y_1Y_2$  is less than  $Y_A Y_B$  and having a second value if  $Y_1Y_2$  is equal to or greater than  $Y_A Y_B$ ;  
 20 and  
 reformatting the symbolic representation of the date with the values  $C_1C_2$ ,  $Y_1Y_2$ ,  $M_1M_2$ , and  $D_1D_2$  to facilitate further processing of the dates.
- 25 2. The method of claim 1, wherein the 10-decade window includes the decade beginning in the year 2000.
3. The method of claim 2, wherein the step of determining includes the step of  
 determining the first value as 20 and the second value as  
 30 19.
4. The method of claim 1, including an additional step, after the step of reformatting, of  
 sorting the symbolic representations of dates.
5. The method of claim 1, wherein the step of reformatting  
 35 includes the step of  
 reformatting each symbolic representation of a date into the format  $C_1C_2Y_1Y_2M_1M_2D_1D_2$ .
6. The method of claim 5, including an additional step, after the step of reformatting, of  
 40 sorting the symbolic representations of dates using a numerical-order sort.
7. The method of claim 1, wherein the step of providing a database includes the step of  
 converting pre-existing date information having a different format into the format wherein  $M_1M_2$  is the numerical month designator,  $D_1D_2$  is the numerical day designator and  $Y_1Y_2$  is the numerical year designator.
8. The method of claim 1, wherein the step of selecting  
 includes the step of  
 50 selecting  $Y_A Y_B$  such that  $Y_B$  is 0 (zero).
9. The method of claim 1, including an additional step, after the step of reformatting, of  
 storing the symbolic representation of dates and their associated information back into the database.
- 55 10. The method of claim 9, including the additional step, after the step of reformatting, of  
 manipulating information in the database having the reformatted date information therein.
- 60 11. A method of processing dates in a database, comprising the steps of  
 providing a database with dates stored therein according to a format wherein  $M_1M_2$  is the numerical month designator,  $D_1D_2$  is the numerical day designator, and  
 65  $Y_1Y_2$  is the numerical year designator, all of dates falling within a 10-decade period of time which includes the decade beginning in the year 2000;

selecting a 10-decade window with a  $Y_A Y_B$  value for the first decade of the window,  $Y_A Y_B$  being no later than the earliest  $Y_1 Y_2$  year designator in the database;

determining a century designator  $C_1 C_2$  for each date in the database,  $C_1 C_2$  having a first value if  $Y_1 Y_2$  is less than  $Y_A Y_B$  and having a second value if  $Y_1 Y_2$  is equal to or greater than  $Y_A Y_B$ ;

reformatting each date in the form  $C_1 C_2 Y_1 Y_2 M_1 M_2 D_1 D_2$  to facilitate further processing of the dates; and

sorting the dates in the form  $C_1 C_2 Y_1 Y_2 M_1 M_2 D_1 D_2$ .

12. The method of claim 11, wherein the step of providing a database includes the step of

converting pre-existing date information having a different format into the format wherein  $M_1 M_2$  is the numeri-

cal month designator,  $D_1D_2$  is the numerical day designator and  $Y_1Y_2$  is the numerical year designator.

13. The method of claim 11, wherein the step of selecting includes the step of

5 selecting  $Y_A Y_B$  such that  $Y_B$  is 0 (zero).

14. The method of claim 11, including an additional step, after the step of sorting, of

storing the sorted dates and their associated information back into the database.

10 15. The method of claim 14, including the additional step, after the step of sorting, of

manipulating information in the database having the reformatted date therein.

16. (New) A method of processing symbolic representations of dates stored in a database, comprising the steps of:

- 5     providing a database with symbolic representations of dates stored therein according to a format wherein  $M_1$   $M_2$  is the numerical month designator,  $D_1$   $D_2$  is the numerical day designator, and  $Y_1$   $Y_2$  is the numerical year designator, all of the symbolic representations
- 10     of dates falling within a 10-decade period of time; selecting a window with a  $Y_A$   $Y_B$  value for a pivot date of the window,  $Y_A$   $Y_B$  being no later than the earliest  $Y_1$   $Y_2$  year designator in the database;
- 15     determining a century designator  $C_1$   $C_2$  for each symbolic representation of a date in the database,  $C_1$   $C_2$  having a first value if  $Y_1$   $Y_2$  is less than  $Y_A$   $Y_B$  and having a second value if  $Y_1$   $Y_2$  is equal to or greater than  $Y_A$   $Y_B$  ; and
- 20     reformatting the symbolic representation of each symbolic representation of a date in the database, without the addition of any new data field to the database, with the reformatted symbolic representation of each date in the database having the values  $C_1$   $C_2$ ,  $Y_1$   $Y_2$ ,  $M_1$   $M_2$ , and  $D_1$   $D_2$ , in order to

facilitate collectively further processing the  
reformatted symbolic representations of each of the  
symbolic representations of each of the dates.

17. (New) The method of claim 16, wherein the window  
5 includes at least a portion of the decade beginning in  
the year 2000.

18. (New) The method of claim 17, wherein the step of  
determining includes the step of:

10 determining the first value as 20 and the second  
value as 19.

19. (New) The method of claim 16, including an  
additional step, after the step of reformatting, of:  
sorting the symbolic representations of dates.

15 20. (New) The method of claim 16, wherein the step of  
reformatting includes the step of:  
reformatting each symbolic representation of a date  
into the format C<sub>1</sub> C<sub>2</sub> Y<sub>1</sub> Y<sub>2</sub> M<sub>1</sub> M<sub>2</sub> D<sub>1</sub> D<sub>2</sub> separately from  
the symbolic representations in the database.

20 21. (New) The method of claim 20, including an  
additional step, after the step of reformatting, of:  
sorting the symbolic representations of dates using a  
numerical-order sort.



22. (New) The method of claim 16, wherein the step of providing a database includes the step of:

5     converting pre-existing date information having a different format into the format wherein  $M_1 M_2$  is the numerical month designator,  $D_1 D_2$  is the numerical day designator and  $Y_1 Y_2$  is the numerical year designator.

23. (New) The method of claim 16, wherein the step of selecting includes the step of:

10     selecting  $Y_A Y_B$  such that  $Y_B$  is 0 (zero).

24. (New) The method of claim 16, including an additional step, after the step of reformatting, of:

storing the symbolic representation of dates and their associated information back into the database.

15 25. (New) The method of claim 24, including the additional step, after the step of reformatting, of:

manipulating information in the database having the reformatted date information therein.

20 26. (New) A method of processing dates in a database, comprising the steps of:

providing a database with dates stored therein according to a format wherein  $M_1 M_2$  is the numerical month designator,  $D_1 D_2$  is the numerical day

designator, and  $Y_1 Y_2$  is the numerical year  
designator, all of the symbolic representations of  
dates falling within a 10-decade period of time;  
selecting a window with a  $Y_A Y_B$  value for a pivot  
5 date of the window,  $Y_A Y_B$  being no later than the  
earliest  $Y_1 Y_2$  year designator in the database;  
determining a century designator  $C_1 C_2$  for each date  
in the database,  $C_1 C_2$  having a first value if  $Y_1 Y_2$   
is less than  $Y_A Y_B$  and having a second value if  $Y_1 Y_2$   
10 is equal to or greater than  $Y_A Y_B$  ;  
reformatting the symbolic representation of each  
symbolic representation of a date in the database,  
without the addition of any new data field to the  
database, with the reformatted symbolic  
15 representation of each date in the database having  
the values  $C_1 C_2, Y_1 Y_2, M_1 M_2$  , and  $D_1 D_2$ , in order to  
facilitate collectively further processing the  
reformatted symbolic representations of each of the  
symbolic representations of each of the dates; and  
20 sorting the dates in the form  $C_1 C_2 Y_1 Y_2 M_1 M_2 D_1 D_2$ .

27. (New) The method of claim 26, wherein the step of  
providing a database includes the step of:  
converting pre-existing date information having a  
different format into the format wherein  $M_1 M_2$  is the

numerical month designator, D<sub>1</sub> D<sub>2</sub> is the numerical  
day designator and Y<sub>1</sub> Y<sub>2</sub> is the numerical year  
designator.

28. (New) The method of claim 26, wherein the step of  
5 selecting includes the step of:

selecting Y<sub>A</sub> Y<sub>B</sub> such that Y<sub>B</sub> is 0 (zero).

29. (New) The method of claim 26, including an  
additional step, after the step of sorting, of:  
storing the sorted dates and their associated  
10 information back into the database.

30. (New) The method of claim 29, including the  
additional step, after the step of sorting, of:

manipulating information in the database having the  
reformatted dates therein.

31. (New) A method of processing symbolic  
15 representations of dates stored in a database,  
comprising the steps of:

providing a database with symbolic representations of  
dates stored therein according to a format wherein Y<sub>1</sub>  
20 Y<sub>2</sub> is the numerical year designator;

selecting a window with a Y<sub>A</sub> Y<sub>B</sub> value for the first  
decade of the window, Y<sub>A</sub> Y<sub>B</sub> being no later than the  
earliest Y<sub>1</sub> Y<sub>2</sub> year designator in the database;

determining a century designator  $C_1$   $C_2$  for each  
symbolic representation of a date in the database,  $C_1$   
 $C_2$  having a first value if  $Y_1$   $Y_2$  is less than  $Y_A$   $Y_B$   
and having a second value if  $Y_1$   $Y_2$  is equal to or  
5 greater than  $Y_A$   $Y_B$  ; and

reformatting the symbolic representation of each  
symbolic representation of a date in the database,  
without the addition of any new data field to the  
database, with the reformatted symbolic  
10 representation of each date in the database having  
the values  $C_1$   $C_2$ ,  $Y_1$   $Y_2$ , in order to facilitate  
collectively further processing the reformatted  
symbolic representations of each of the symbolic  
representations of each of the dates.

15 32. (New) A method of processing dates in a database,  
comprising the steps of:

providing a database with symbolic representations of  
dates stored therein according to a format wherein  $Y_1$   
 $Y_2$  is the numerical year designator;

20 selecting a window with a  $Y_A$   $Y_B$  value for a pivot  
year of the window,  $Y_A$   $Y_B$  being no later than the  
earliest  $Y_1$   $Y_2$  year designator in the database;

determining a century designator  $C_1$   $C_2$  for each  
symbolic representation of a date in the database,  $C_1$

C<sub>2</sub> having a first value if Y<sub>1</sub> Y<sub>2</sub> is less than Y<sub>A</sub> Y<sub>B</sub>  
and having a second value if Y<sub>1</sub> Y<sub>2</sub> is equal to or  
greater than Y<sub>A</sub> Y<sub>B</sub> ;

5 reformatting the symbolic representation of each of  
the dates in the database, without the addition of  
any new data field to the database, with the  
reformatted symbolic representation of each date in  
the database having the values C<sub>1</sub> C<sub>2</sub>, Y<sub>1</sub> Y<sub>2</sub>, in order  
10 to facilitate collectively further processing the  
reformatted symbolic representations of each of the  
dates; and

sorting the dates in the form C<sub>1</sub> C<sub>2</sub> Y<sub>1</sub> Y<sub>2</sub>.

33. (New) A method of processing symbolic  
representations of dates stored in a database,  
15 comprising the steps of:

providing a database with symbolic representations of  
dates stored therein according to a format wherein Y<sub>1</sub>  
Y<sub>2</sub> is the numerical year designator;

20 selecting a window with a Y<sub>A</sub> Y<sub>B</sub> value for the first  
decade of the window, Y<sub>A</sub> Y<sub>B</sub> being no later than the  
earliest Y<sub>1</sub> Y<sub>2</sub> year designator in the database;

determining a century designator C<sub>1</sub> C<sub>2</sub> for each  
symbolic representation of a date in the database, C<sub>1</sub>  
C<sub>2</sub> having a first value if Y<sub>1</sub> Y<sub>2</sub> is less than Y<sub>A</sub> Y<sub>B</sub>

and having a second value if  $Y_1 Y_2$  is equal to or  
greater than  $Y_A Y_B$  ; and  
reformatting the symbolic representation of each  
symbolic representation of a date in the database,  
5 without changing any of the symbolic representations  
of a date in the database during the reformatting  
step, with the reformatted symbolic representation of  
each date in the database having the values  $C_1 C_2, Y_1$   
 $Y_2$ , in order to facilitate collectively further  
10 processing the reformatted symbolic representations  
of each of the dates.

34. (New) A method for representing and utilizing dates  
stored in at least one date field of a database  
utilizing symbolic representations of the dates stored  
15 in the at least one date field of the database, which  
are in a format that creates ambiguity between dates in  
each of a pair of adjacent centuries, comprising the  
steps of:

converting each of the symbolic representations of  
20 dates stored in the at least one date field of the  
database to a symbolic representation of each of the  
respective dates that does not create the ambiguity,  
by windowing the symbolic representations of each of  
the respective dates as stored in the at least one  
25 date field of the database against a pivot year

represented by one of the symbolic representations of  
the dates as stored in the at least one date field of  
the database, without the addition of any new data  
field to the database for purposes of such windowing  
5 and converting; and,

running a program collectively on each of the  
converted symbolic representations of each of the  
respective dates to sort or otherwise manipulate the  
dates represented by the converted symbolic  
10 representations, separately from the date data  
symbolic representations contained in the at least  
one date field of the database.

35. (New) A method of claim 34 further comprising the  
step of:  
15 opening the database prior to the step of  
converting.

36. (New) The method of claim 34 further comprising  
the step of:  
20 collectively sorting the converted symbolic  
representations prior to the step of running the  
program on the converted symbolic representations.

37. (New) The method of claim 35 further comprising  
25 the step of:

collectively sorting the converted symbolic representations prior to the step of running the program on the converted symbolic representations.

5    38. (New) The method of claim 34 further comprising the step of:

collectively manipulating the converted symbolic representations prior to the step of running the program on the converted symbolic representations.

10

39. (New) The method of claim 35 further comprising the step of:

collectively manipulating the converted symbolic representations prior to the step of running the program on the converted symbolic representations.

15

40. (New) The method of claim 34 further comprising the step of:

collectively sorting the converted symbolic representations according to a different data field contained in the database from the at least one date field, prior to the step of running the program on the converted symbolic representations.

20



41. (New) The method of claim 35 further comprising  
the step of:

collectively sorting the converted symbolic  
representations according to a different data field  
5 contained in the database from the at least one date  
field, prior to the step of running the program on the  
converted symbolic representations.

42. (New) The method of claim 34 further comprising  
10 the step of:

collectively manipulating the converted symbolic  
representations according to a different data field  
contained in the database from the at least one date  
field, prior to the step of running the program on the  
15 converted symbolic representations.

43. (New) The method of claim 35 further comprising  
the step of:

collectively manipulating the converted symbolic  
20 representations according to a different data entry  
field contained in the database from the at least one  
date field, prior to the step of running the program on  
the converted symbolic representations.

44. (New) The method of claim 34 wherein the program performs an operation which manipulates the data in a data field associated with the at least one date field of the database according to the converted symbolic representation of the date.

5

45. (New) The method of claim 35 wherein the program performs an operation which manipulates the data in a data field associated with the at least one date field of the database according to the converted symbolic representation of the date.

10

46. (New) The method of claim 34 wherein the step of converting includes converting at least a substantial portion of each of the plurality of symbolic representations of dates in the at least one date field and repeating this step until each of the date data entries in the at least one date field is converted into the format that does not have the ambiguity.

15

20

47. (New) The method of claim 35 wherein the step of converting includes converting at least a substantial portion of each of the plurality of symbolic representations of dates in the at least one date field and repeating this step until each of the date data

25

entries in the at least one date field is converted  
into the format that does not have the ambiguity.

48. (New) The method of claim 46 further comprising  
5 the steps of:

collectively sorting the converted symbolic  
representations prior to the step of running the  
program on the converted symbolic representations.

10 49. (New) The method of claim 47 further comprising  
the steps of:

collectively sorting the converted symbolic  
representations prior to the step of running the  
program on the converted symbolic representations.

15 50. (New) The method of claim 46 further comprising  
the step of:

collectively manipulating the converted symbolic  
representations.

20 51. (New) The method of claim 49 further comprising  
the step of:

collectively manipulating the converted symbolic  
representations.

25

52. (New) The method of claim 46 further comprising  
the step of:

collectively sorting the converted symbolic  
representations according to a different data field in  
5 the database than the at least one date field, prior to  
the step of running the program.

53. (New) The method of claim 47 further comprising  
the step of:

10 collectively sorting the converted symbolic  
representations according to a different data field in  
the database than the at least one date field, prior to  
the step of running the program.

15 54. (New) The method of claim 52 further comprising  
the step of:

collectively manipulating the converted symbolic.

55. (New) The method of claim 53 further comprising  
20 the step of:

collectively manipulating the converted symbolic  
representations.

56. (New) The method of claim 52 wherein the program  
25 performs an operation which manipulates the data in a

data field associated with the at least one date field  
of the database according to the converted symbolic  
representation of the date.

5    57. (New) The method of claim 53 wherein the program  
performs an operation which manipulates the data in a  
data field associated with the at least one date field  
of the database according to the converted symbolic  
representation of the date.

10    58. (New) The method of claim 54 wherein the program  
performs an operation which manipulates the data in a  
data field associated with the at least one date field  
of the database according to the converted symbolic  
15    representation of the date.

59. (New) The method of claim 55 wherein the program  
performs an operation which manipulates the data in a  
data field associated with the at least one date field  
20    of the database according to the converted symbolic  
representation of the date.

60. (New) A method for representing and utilizing dates  
stored in at least one date field of a database  
25    utilizing symbolic representations of the dates stored

in the at least one date field of the database, which  
are in a format that creates ambiguity between dates in  
each of a pair of adjacent centuries, comprising the  
steps of:

- 5     converting each of the symbolic representations of  
      dates stored in the at least one date field of the  
      database to a symbolic representation of each of the  
      respective dates that does not create the ambiguity,  
      by windowing the symbolic representations of each of
- 10    the respective dates as stored in the at least one  
      date field of the database against a pivot year  
      represented by one of the symbolic representations of  
      the dates as stored in the at least one date field of  
      the database, without modifying any of the symbolic
- 15    representations of dates in the at least one date  
      field of the database for purposes of such windowing  
      and converting;
- running a program on each of the converted symbolic  
      representations of each of the respective dates to
- 20    sort or otherwise manipulate data in the database  
      according to the dates represented by the converted  
      symbolic representations, separately from the date  
      data symbolic representations of dates contained in  
      the at least one date field of the database.

61. (New) A method for representing and utilizing dates  
stored in at least one date field of a database  
utilizing symbolic representations of the dates stored  
in the at least one date field of the database, which  
5 are in a format that creates ambiguity between dates in  
each of a pair of adjacent centuries, comprising the  
steps of:

converting each of the symbolic representations of  
dates stored in the at least one date field of the  
10 database to a symbolic representation of each of the  
respective dates that does not create the ambiguity,  
by windowing the symbolic representations of each of  
the respective dates as stored in the at least one  
date field of the database against a pivot year  
15 represented by one of the symbolic representations of  
the dates as stored in the at least one date field of  
the database, without modifying any of the symbolic  
representations of dates in the at least date field  
of the database for purposes of such windowing and  
20 converting;

running a program collectively on each of the  
converted symbolic representations of each of the  
respective dates to sort or otherwise manipulate the  
dates represented by the converted symbolic  
25 representations, separately from the symbolic

representations of dates contained in the at least  
one date field of the database.

62. (New) A method for representing and utilizing dates  
stored in at least one date field of a database

5 utilizing symbolic representations of the dates stored  
in the at least one date field of the database, which  
are in a format that creates ambiguity between dates in  
each of a pair of adjacent centuries, comprising the  
steps of:

10 converting each of the symbolic representations of  
dates stored in the at least one date field of the  
database to a symbolic representation of each of the  
respective dates that does not create the ambiguity,  
by windowing the symbolic representations of each of  
15 the respective dates as stored in the at least one  
date field of the database against a pivot year  
represented by one of the symbolic representations of  
the dates as stored in the at least one date field of  
the database, without the addition of any new data  
20 field to the database for purposes of such windowing  
and converting;

storing the converted symbolic representations  
separate from the at least one date field of the  
database; and



running a program on the stored converted symbolic  
representations to sort or otherwise manipulate data  
in the database according to the dates represented by  
the converted symbolic representations, separately  
5 from the symbolic representations of dates contained  
in the at least one date field of the database.

63. (New) A method for representing and utilizing dates  
stored in at least one date field of a database  
utilizing symbolic representations of the dates stored  
10 in the at least one date field of the database, which  
are in a format that creates ambiguity between dates in  
each of a pair of adjacent centuries, comprising the  
steps of:

converting each of the symbolic representations of  
15 dates stored in the at least one date field of the  
database to a symbolic representation of each of the  
respective dates that does not create the ambiguity,  
by windowing the symbolic representations of each of  
the respective dates as stored in the at least one  
20 date field of the database against a pivot year  
represented by one of the symbolic representations of  
the dates as stored in the at least one date field of  
the database, without the addition of any new data  
field to the database for purposes of such windowing  
25 and converting;

storing the converted symbolic representations  
separate from the at least one date field of the  
database; and

5 running a program collectively on the stored  
converted symbolic representations to sort or  
otherwise manipulate the dates represented by the  
converted symbolic representations, separately from  
the symbolic representations of dates contained in  
the at least one date field of the database.

10

64. (New) A method for representing and utilizing dates  
stored in at least one date field of a database  
utilizing symbolic representations of the dates stored  
in the at least one date field of the database, which  
15 are in a format that creates ambiguity between dates in  
each of a pair of adjacent centuries, comprising the  
steps of:

converting each of the symbolic representations of  
dates stored in the at least one date field of the  
20 database to a symbolic representation of each of the  
respective dates that does not create the ambiguity,  
by windowing the symbolic representations of each of  
the respective dates as stored in the at least one  
date field of the database against a pivot year  
25 represented by one of the symbolic representations of

the dates as stored in the at least one date field of  
the database, without modifying any of the symbolic  
representations of dates in the at least one date  
field of the database for purposes of such windowing  
and converting;

storing the converted symbolic representations  
separate from the at least one date field in the  
database; and

running a program on the stored converted symbolic  
representations to sort or otherwise manipulate data  
in the database according to the dates represented by  
the converted symbolic representations, separately  
from the symbolic representations of dates contained  
in the at least one date field of the database.

65. (New) A method for representing and utilizing dates  
stored in at least one date field of a database  
utilizing symbolic representations of the dates stored  
in the at least one date field of the database, which  
are in a format that creates ambiguity between dates in  
each of a pair of adjacent centuries, comprising the  
steps of:

converting each of the symbolic representations of  
dates stored in the at least one date field of the  
database to a symbolic representation of each of the  
respective dates that does not create the ambiguity,

- by windowing the symbolic representations of each of  
the respective dates as stored in the at least one  
date field of the database against a pivot year  
represented by one of the symbolic representations of  
5 the dates as stored in the at least one date field of  
the database, without modifying any of the symbolic  
representations of dates in the at least one date  
field of the database for purposes of such windowing  
and converting;
- 10 storing the converted symbolic representations  
separate from the at least one date field in the  
database; and
- running a program collectively on the stored  
converted symbolic representations to sort or  
15 otherwise manipulate the dates represented by the  
converted symbolic representations, separately from  
the symbolic representations of dates contained in  
the at least one date field of the database.
- 20 66. (New) A method of processing dates in a database,  
comprising the steps of:
- providing a database with dates stored in at least  
one date field therein according to a format wherein  
M<sub>1</sub> M<sub>2</sub> is the numerical month designator, D<sub>1</sub> D<sub>2</sub> is the

numerical day designator, and  $Y_1 Y_2$  is the numerical year designator;

selecting a window with a  $Y_A Y_B$  value for a pivot date of the window,  $Y_A Y_B$  being no later than the  
5 earliest  $Y_1 Y_2$  year designator in the database;

determining a century designator  $C_1 C_2$  for each date in the database,  $C_1 C_2$  having a first value if  $Y_1 Y_2$  is less than  $Y_A Y_B$  and having a second value if  $Y_1 Y_2$  is equal to or greater than  $Y_A Y_B$ ;

10 reformatting the symbolic representation of each symbolic representation of a date in a portion of the at least one date field in the database, without the addition of any new data field to the database, with the reformatted symbolic representation of each date  
15 in the database having the values  $C_1 C_2$ ,  $Y_1 Y_2$ ,  $M_1 M_2$ , and  $D_1 D_2$ ; and

repeating the step of reformatting until each symbolic representation of a date in the at least one date field has been reformatted in order to  
20 facilitate collectively further processing the reformatted symbolic representations of each of the symbolic representations of each of the dates.

67. (New) A method of processing dates in a database, comprising the steps of:

providing a database with dates stored in at least  
one date field therein according to a format wherein  
Y<sub>1</sub> Y<sub>2</sub> is the numerical year designator;

5 selecting a window with a Y<sub>A</sub> Y<sub>B</sub> value for a pivot  
date of the window, Y<sub>A</sub> Y<sub>B</sub> being no later than the  
earliest Y<sub>1</sub> Y<sub>2</sub> year designator in the database;

10 determining a century designator C<sub>1</sub> C<sub>2</sub> for each date  
in the database, C<sub>1</sub> C<sub>2</sub> having a first value if Y<sub>1</sub> Y<sub>2</sub>  
is less than Y<sub>A</sub> Y<sub>B</sub> and having a second value if Y<sub>1</sub> Y<sub>2</sub>  
is equal to or greater than Y<sub>A</sub> Y<sub>B</sub>;

15 reformatting the symbolic representation of each  
symbolic representation of a date in a portion of the  
at least one date field in the database, without the  
addition of any new data field to the database, with  
the reformatted symbolic representation of each date  
in the database having the values C<sub>1</sub> C<sub>2</sub>, Y<sub>1</sub> Y<sub>2</sub>; and  
repeating the step of reformatting until each  
symbolic representation of a date in the at least one  
date field has been reformatted in order to  
20 facilitate collectively further processing the  
reformatted symbolic representations of each of the  
symbolic representations of each of the dates.

68. (New) A method of processing symbolic  
representations of dates stored in a database,  
comprising the steps of:

providing a database with symbolic representations of  
5 dates stored in at least one date field therein  
according to a format wherein  $Y_1 Y_2$  is the numerical  
year designator;

selecting a window with a  $Y_A Y_B$  value for the first  
decade of the window,  $Y_A Y_B$  being no later than the  
10 earliest  $Y_1 Y_2$  year designator in the at least one  
date field of the database;

determining a century designator  $C_1 C_2$  for each  
symbolic representation of a date in the database,  $C_1$   
 $C_2$  having a first value if  $Y_1 Y_2$  is less than  $Y_A Y_B$   
15 and having a second value if  $Y_1 Y_2$  is equal to or  
greater than  $Y_A Y_B$  ; and

reformatting the symbolic representation of each  
symbolic representation of a date in at least one  
date field in the database, without the addition of  
20 any new data field to the database, with the  
reformatted symbolic representation of each date in  
the database having the values  $C_1 C_2, Y_1 Y_2$ , in order  
to facilitate further processing of the reformatted  
symbolic representations of each of the symbolic

representations of each of the dates, by running a  
program on the reformatted symbolic representations  
of each of the dates.

69. (New) A method of processing dates in a database,  
5 comprising the steps of:
- providing a database with dates stored in at least  
one date field therein according to a format wherein  
Y<sub>1</sub> Y<sub>2</sub> is the numerical year designator;
- 10 selecting a window with a Y<sub>A</sub> Y<sub>B</sub> value for a pivot  
year of the window, Y<sub>A</sub> Y<sub>B</sub> being no later than the  
earliest Y<sub>1</sub> Y<sub>2</sub> year designator in the database;
- determining a century designator C<sub>1</sub> C<sub>2</sub> for each date  
in the at least one date field of the database, C<sub>1</sub> C<sub>2</sub>  
having a first value if Y<sub>1</sub> Y<sub>2</sub> is less than Y<sub>A</sub> Y<sub>B</sub> and  
15 having a second value if Y<sub>1</sub> Y<sub>2</sub> is equal to or greater  
than Y<sub>A</sub> Y<sub>B</sub> ;
- reformatting the symbolic representation of each  
symbolic representation of a date in the at least one  
date field in the database, without the addition of  
20 any new data field to the database, with the  
reformatted symbolic representation of each date in  
the database having the values C<sub>1</sub> C<sub>2</sub>, Y<sub>1</sub> Y<sub>2</sub>;
- sorting the reformatted symbolic representations of  
the dates in the form C<sub>1</sub> C<sub>2</sub> Y<sub>1</sub> Y<sub>2</sub>; and



running a program on the reformatted symbolic  
representations of each of the dates.

70. (New) A method for representing and utilizing dates  
stored in at least one date field of a database
- 5 utilizing symbolic representations of the dates stored  
in at least one date field of the database, which are  
in a format that creates ambiguity between dates in  
each of a pair of adjacent centuries, comprising the  
steps of
- 10 converting each of the symbolic representations of  
dates stored in the at least one date field of the  
database to a symbolic representation of each of the  
respective dates that does not create the ambiguity,  
by windowing the symbolic representations of each of
- 15 the respective dates as stored in the at least one  
date field of the database against a pivot year, with  
the pivot year being less than or equal to the  
earliest date represented by the symbolic  
representation of dates stored in the at least one
- 20 date field, without the addition of any new data  
field to the database, and without modifying any of  
the symbolic representations of dates in the at least  
one date field, for purposes of such windowing and  
converting; and,

running a program on the converted symbolic  
representations of each of the dates to sort or  
otherwise manipulate the dates represented by the  
converted symbolic representations, separately from  
5 the date data symbolic representations contained in  
the at least one date field of the database.

71. (New) A method for representing and utilizing dates  
stored in at least one date field of the database  
utilizing symbolic representations of the dates stored  
10 in the at least one date field of the database, which  
are in a format that creates ambiguity between dates in  
each of a pair of adjacent centuries, comprising the  
steps of

converting each of the symbolic representations of  
15 dates stored in the at least one date field of the  
database to a symbolic representation of each of the  
respective dates that does not create the ambiguity,  
by windowing the symbolic representations of each of  
the respective dates as stored in the at least one  
20 date field of the database against a pivot year, with  
the pivot year being less than or equal to the  
earliest date represented by a symbolic  
representation of dates stored in the at least one  
date field, and without the addition of any new data

field to the database for purposes of such windowing  
and converting;

storing each of the converted symbolic  
representations of each of the dates separate from  
5 the database; and,

running a program on the stored converted symbolic  
representations of each of the converted symbolic  
representations of the dates to sort or otherwise  
manipulate the dates represented by the converted  
10 symbolic representations, separately from the date  
data symbolic representations contained in the at  
least one date field of the database.

72. (New) A method of processing symbolic  
representations of dates stored in a database,

15 comprising the steps of

selecting a database with symbolic representations of  
dates stored therein according to a format wherein  $M_1$   
 $M_2$  is the numerical month designator,  $D_1$   $D_2$  is the  
numerical day designator, and  $Y_1$   $Y_2$  is the numerical  
20 year designator;

selecting a 10-decade window with a  $Y_A$   $Y_B$  value for  
the first decade of the window,  $Y_A$   $Y_B$  being no later  
than the earliest  $Y_1$   $Y_2$  year designator in the  
database;

determining a century designator  $C_1$   $C_2$  for each  
 symbolic representation of a date in the database,  $C_1$   
 $C_2$  having a first value if  $Y_1$   $Y_2$  is less than  $Y_A$   $Y_B$   
 and having a second value if  $Y_1$   $Y_2$  is equal to or  
 5 greater than  $Y_A$   $Y_B$  ; and,

reformatting the symbolic representation of each  
 symbolic representation of a date in the database  
 with the values  $C_1$   $C_2$ ,  $Y_1$   $Y_2$ ,  $M_1$   $M_2$  , and  $D_1$   $D_2$  prior  
 to collectively further processing information  
 10 contained within the database associated with the  
respective dates.

73. (New) A method of processing symbolic  
representations of dates stored in a database,  
comprising the steps of

15 providing a database with symbolic representations of  
dates stored therein according to a format wherein  $Y_1$   
 $Y_2$  is the numerical year designator, all of the  
symbolic representations of dates falling within a  
10-decade period of time;

20 selecting a 10-decade window with a  $Y_A$   $Y_B$  value for  
the first decade of the window,  $Y_A$   $Y_B$  being no later  
than the earliest  $Y_1$   $Y_2$  year designator in the  
database;

determining a century designator  $C_1$   $C_2$  for each  
symbolic representation of a date in the database,  $C_1$   
 $C_2$  having a first value if  $Y_1$   $Y_2$  is less than  $Y_A$   $Y_B$   
and having a second value if  $Y_1$   $Y_2$  is equal to or  
5 greater than  $Y_A$   $Y_B$  ; and,

reformatting the symbolic representation of the date  
with the values  $C_1$   $C_2$ ,  $Y_1$   $Y_2$ , to facilitate further  
processing of the dates.

74. (New) A method of processing dates in a database,  
10 comprising the steps of

providing a database with symbolic representations of  
dates stored therein according to a format wherein  $Y_1$   
 $Y_2$  is the numerical year designator, all of symbolic  
representations of dates falling within a 10-decade  
15 period of time;

selecting a 10-decade window with a  $Y_A$   $Y_B$  value for  
the first decade of the window,  $Y_A$   $Y_B$  being no later  
than the earliest  $Y_1$   $Y_2$  year designator in the  
database;

20 determining a century designator  $C_1$   $C_2$  for each date  
in the database,  $C_1$   $C_2$  having a first value if  $Y_1$   $Y_2$   
is less than  $Y_A$   $Y_B$  and having a second value if  $Y_1$   $Y_2$   
is equal to or greater than  $Y_A$   $Y_B$  ;

reformatting each date in the form C<sub>1</sub> C<sub>2</sub> Y<sub>1</sub> Y<sub>2</sub> to  
facilitate further processing of the dates; and,  
sorting the dates in the form C<sub>1</sub> C<sub>2</sub> Y<sub>1</sub> Y<sub>2</sub>.

75. (New) A method of processing symbolic  
5 representations of dates stored in a database,  
comprising the steps of
- 10 providing a database with symbolic representations of  
dates stored therein according to a format wherein M<sub>1</sub>  
M<sub>2</sub> is the numerical month designator, D<sub>1</sub> D<sub>2</sub> is the  
numerical day designator, and Y<sub>1</sub> Y<sub>2</sub> is the numerical  
year designator;
- selecting a window with a Y<sub>A</sub> Y<sub>B</sub> value for a pivot  
date of the window, Y<sub>A</sub> Y<sub>B</sub> being no later than the  
earliest Y<sub>1</sub> Y<sub>2</sub> year designator in the database;
- 15 determining a century designator C<sub>1</sub> C<sub>2</sub> for each  
symbolic representation of a date in the database, C<sub>1</sub>  
C<sub>2</sub> having a first value if Y<sub>1</sub> Y<sub>2</sub> is less than Y<sub>A</sub> Y<sub>B</sub>  
and having a second value if Y<sub>1</sub> Y<sub>2</sub> is equal to or  
greater than Y<sub>A</sub> Y<sub>B</sub> ; and
- 20 reformatting the symbolic representation of each  
symbolic representation of a date in the database,  
without the addition of any new data field to the  
database, with the reformatted symbolic  
representation of each date in the database having

the values C<sub>1</sub> C<sub>2</sub>, Y<sub>1</sub> Y<sub>2</sub>, M<sub>1</sub> M<sub>2</sub>, and D<sub>1</sub> D<sub>2</sub>, in order to  
facilitate further processing of the reformatted  
symbolic representations of each of the symbolic  
representations of each of the dates.

- 5    76. (New) A method of processing dates in a database,  
comprising the steps of

providing a database with dates stored therein  
according to a format wherein M<sub>1</sub> M<sub>2</sub> is the numerical  
month designator, D<sub>1</sub> D<sub>2</sub> is the numerical day  
10    designator, and Y<sub>1</sub> Y<sub>2</sub> is the numerical year  
designator;

selecting a window with a Y<sub>A</sub> Y<sub>B</sub> value for a pivot  
date of the window, Y<sub>A</sub> Y<sub>B</sub> being no later than the  
earliest Y<sub>1</sub> Y<sub>2</sub> year designator in the database;

- 15    determining a century designator C<sub>1</sub> C<sub>2</sub> for each date  
in the database, C<sub>1</sub> C<sub>2</sub> having a first value if Y<sub>1</sub> Y<sub>2</sub>  
is less than Y<sub>A</sub> Y<sub>B</sub> and having a second value if Y<sub>1</sub> Y<sub>2</sub>  
is equal to or greater than Y<sub>A</sub> Y<sub>B</sub> ;

reformatting the symbolic representation of each  
20    symbolic representation of a date in the database,  
without the addition of any new data field to the  
database, with the reformatted symbolic  
representation of each date in the database having  
the values C<sub>1</sub> C<sub>2</sub>, Y<sub>1</sub> Y<sub>2</sub>, M<sub>1</sub> M<sub>2</sub> , and D<sub>1</sub> D<sub>2</sub>, in order to

facilitate further processing of the reformatted  
symbolic representations of each of the symbolic  
representations of each of the dates; and  
sorting the dates in the form C<sub>1</sub> C<sub>2</sub> Y<sub>1</sub> Y<sub>2</sub> M<sub>1</sub> M<sub>2</sub> D<sub>1</sub> D<sub>2</sub>.



[11] Patent Number: 5,806,063

[45] Date of Patent: Sep. 8, 1998

#### OTHER PUBLICATIONS

The Year 2000 and 2-Digit Dates: A Guide for Planning and Implementation, Third Edition, May, 1996.

IBM: *The Year 2000 and 2-Digit Dates: A Guide for Planning and Implementation*; First Edition, Oct. 1995.

Primary Examiner—Wayne Amsbury

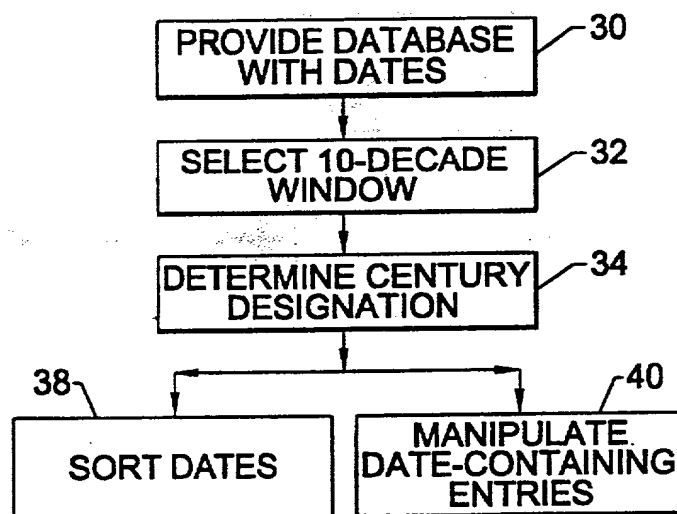
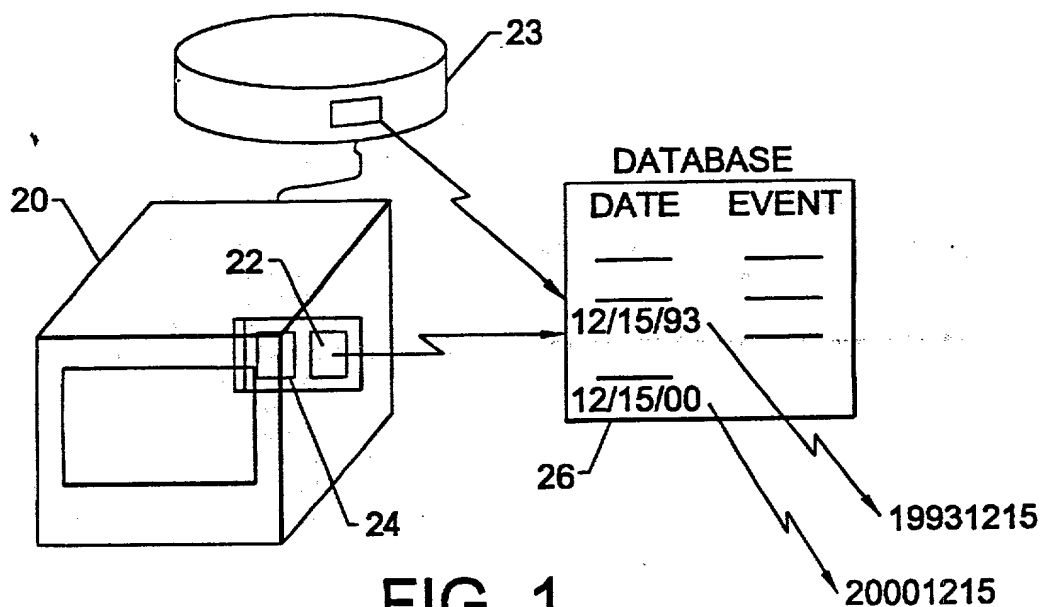
Attorney, Agent, or Firm—Bell Seltzer Intellectual Property Group of Alston & Bird LLP

[57]

#### ABSTRACT

Dates stored in symbolic form in a database are reformatted to permit easy manipulation and sorting of date-related information. Each date in  $M_1M_2$ ,  $D_1D_2$ , and  $Y_1Y_2$  format is converted to  $C_1C_2$ ,  $Y_1Y_2$ ,  $M_1M_2$ , and  $D_1D_2$  format. To accomplish the conversion, a 10-decade window starting on  $Y_A Y_B$  is defined that encompasses all dates in the database. The value of  $C_1C_2$  is determined by the relative values of  $Y_1Y_2$  and  $Y_A Y_B$ . The reformatted date information is particularly useful when the reformatting is in  $C_1C_2Y_1Y_2M_1M_2D_1D_2$  format, because sorting by date is accomplished using a pure numerical-value sort.

15 Claims, 1 Drawing Sheet



Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

<b>REISSUE APPLICATION DECLARATION BY THE ASSIGNEE</b>		Docket Number (optional)
<p>I hereby declare that:</p> <p>My residence and post office address and citizenship are stated below next to my name.</p> <p>I am authorized to act on behalf of the following assignee: <u>DICKENS-SOEDER2000</u></p> <p>and the title of my position with said assignee is: <u>Manager/Member</u></p> <p>The entire title to the patent identified below is vested in said assignee.</p>		
Name of Patentee(s): <u>Bruce Dickens</u>		
Patent Number <u>5,806,063</u>	Date of Patent Issued	
Title of Invention <u>Date Formatting and Sorting for Dates Spanning the Turn of the Century</u>		
<p>I believe said patentee(s) to be the original, first and sole/joint inventor(s) of the subject matter which is described and claimed in said patent, for which a reissue patent is sought on the invention entitled <u>Date Formatting and Sorting For Dates Spanning the Turn of the Century</u>, the specification of which</p> <p><input checked="" type="checkbox"/> is attached hereto.</p> <p><input type="checkbox"/> was filed on _____ as reissue application number _____ / _____ and was amended on _____ (If applicable)</p> <p>I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment referred to above.</p> <p>I acknowledge the duty to disclose information which is material to patentability as defined in 37 CFR 1.56.</p> <p>I verily believe the original patent to be wholly or partly inoperative or invalid, for the reasons described below. (Check all boxes that apply.)</p> <p><input type="checkbox"/> by reason of a defective specification or drawing.</p> <p><input checked="" type="checkbox"/> by reason of the patentee claiming more or less than he had the right to claim in the patent.</p> <p><input type="checkbox"/> by reason of other errors.</p> <p>At least one error upon which reissue is based is described as follows:  <u>The claims through no fault or deception, intent of the inventor are more narrow than allowed by the specification and the prior art.</u></p> <p style="text-align: center;">[Attach additional sheets, if needed.]</p> <p>All errors corrected in this reissue application arose without any deceptive intention on the part of the applicant.</p>		

[Page 1 of 2]

Burden Hour Statement: This form is estimated to take 0.5 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Washington, DC 20231.

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number

## (REISSUE APPLICATION DECLARATION BY THE ASSIGNEE, page 2)

Docket Number (Optional )

I hereby appoint the following attorney(s) and/or agent(s) to prosecute this application and transact all business in the Patent and Trademark Office connected therewith.

Name(s)

William C. Cray

Registration Number

27627

Correspondence Address: Direct all communications about the application to:

☐ Customer Number

022853

Type Customer Number Here

Place Customer  
Number Bar Code  
Label Here

OR

☐ Firm or  
Individual  
Name

Address

Address

City

State

Zip

Country

Telephone

Fax

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine and imprisonment, or both, under 18 U.S.C. 1001, and that such willful false statements may jeopardize the validity of the application, any patent issuing thereon, or any patent to which this declaration is directed.

Full name of person signing (given name, family name)

Bruce M. Dickens

Signature

Bruce Dickens

Date

02/22/00

Address of Assignee

384 Forest Avenue, Suite 13 Laguna Beach, CA 92651

Patentee

Citizenship

Residence/Post Office Address

Patentee

Citizenship

Residence/Post Office Address

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

**STATEMENT CLAIMING SMALL ENTITY STATUS**  
**(37 CFR 1.9(f) & 1.27(c))--SMALL BUSINESS CONCERN**

Docket Number (Optional)

Applicant, Patentee, or Identifier: Dickens  
Application or Patent No.: 5,806,063  
Filed or Issued: September 8, 1998  
Title: Formatting and Sorting for Dates Spanning the Turn of the Century

I hereby state that I am

- ☐ the owner of the small business concern identified below:  
☒ an official of the small business concern empowered to act on behalf of the concern identified below:

NAME OF SMALL BUSINESS CONCERN Dickens-Soeder2000

ADDRESS OF SMALL BUSINESS CONCERN 384 forest Avenue, Suite 13  
Laguna Beach, CA 92651

I hereby state that the above identified small business concern qualifies as a small business concern as defined in 13 CFR Part 121 for purposes of paying reduced fees to the United States Patent and Trademark Office. Questions related to size standards for a small business concern may be directed to: Small Business Administration, Size Standards Staff, 409 Third Street, SW, Washington, DC 20416.

I hereby state that rights under contract or law have been conveyed to and remain with the small business concern identified above with regard to the invention described in:

- ☒ the specification filed herewith with title as listed above.  
☐ the application identified above.  
☐ the patent identified above.

If the rights held by the above identified small business concern are not exclusive, each individual, concern, or organization having rights in the invention must file separate statements as to their status as small entities, and no rights to the invention are held by any person, other than the inventor, who would not qualify as an independent inventor under 37 CFR 1.9(c) if that person made the invention, or by any concern which would not qualify as a small business concern under 37 CFR 1.9(d), or a nonprofit organization under 37 CFR 1.9(e).

- Each person, concern, or organization having any rights in the invention is listed below:  
☒ no such person, concern, or organization exists.  
☐ each such person, concern, or organization is listed below.

Separate statements are required from each named person, concern or organization having rights to the invention stating their status as small entities. (37 CFR 1.27)

I acknowledge the duty to file, in this application or patent, notification of any change in status resulting in loss of entitlement to small entity status prior to paying, or at the time of paying, the earliest of the issue fee or any maintenance fee due after the date on which status as a small entity is no longer appropriate. (37 CFR 1.28(b))

NAME OF PERSON SIGNING Bruce Dickens

TITLE OF PERSON IF OTHER THAN OWNER Member/Manager

ADDRESS OF PERSON SIGNING 3892 Cedron, Irvine, CA 92606

SIGNATURE Bruce Dickens DATE 2/18/00 <sup>22 BD</sup>